MikroTik RouterOS™ v2.9
Reference Manual

Table Of Contents

Device Driver List....1
- General Information....2
- Ethernet....2
- Wireless....9
- Aironet Arlan....11
- RadioLAN....12
- Synchronous Serial....12
- Asynchronous Serial....12
- ISDN....13
- VoIP....13
- xDSL....14
- HomePNA....14
- LCD....14
- PCMCIA Adapters....14
- GPRS Cards....14

License Management....16
- General Information....16
- License Management....18

Specifications Sheet....21
- General Information....21

Basic Setup Guide....25
- General Information....25
- Setting up MikroTik RouterOS™....26
- Logging into the MikroTik Router....29
- Adding Software Packages....30
- Navigating The Terminal Console....30
- Basic Configuration Tasks....33
- Setup Command....34
- Basic Examples....35
- Advanced Configuration Tasks....38

Installing RouterOS with CD-Install....40
- CD-Install....40

Installing RouterOS with Floppies....42
- Floppy Install....42

Installing RouterOS with NetInstall....43
- NetInstall....43

Configuration Management....46
- General Information....46
- System Backup....47
- The Export Command....47
- The Import Command....48
- Configuration Reset....49

FTP (File Transfer Protocol) Server....50

General Information 50
File Transfer Protocol Server..... 50
MAC Level Access (Telnet and Winbox)..... 52
General Information 52
MAC Telnet Server..... 53
MAC WinBox Server..... 53
Monitoring Active Session List..... 54
MAC Telnet Client..... 54
Serial Console and Terminal..... 55
General Information 55
Serial Console Configuration..... 56
Configuring Console..... 56
Using Serial Terminal..... 57
Console Screen..... 58
Software Package Management..... 59
General Information 59
Installation (Upgrade)..... 60
Uninstalling..... 61
Downgrading..... 62
System Upgrade..... 63
Adding Package Source..... 64
Software Package List..... 65
Software Version Management..... 68
General Information 68
System Upgrade..... 68
Adding Package Source..... 70
SSH (Secure Shell) Server and Client..... 71
General Information 71
SSH Server..... 72
SSH Client..... 72
Telnet Server and Client..... 74
General Information 74
Telnet Server..... 74
Telnet Client..... 75
Terminal Console..... 76
General Information 76
Common Console Functions..... 77
Lists and Item Names..... 78
Quick Typing..... 79
Additional Information..... 80
General Commands..... 80
Safe Mode..... 82
Winbox..... 84
General Information..... 84
Troubleshooting..... 85
IP Addresses and ARP..... 87

General Information 87
IP Addressing..... 88
Address Resolution Protocol..... 89
Proxy-ARP feature..... 90
Unnumbered Interfaces..... 91
Troubleshooting..... 92
OSPF..... 93
General Information 93
General Setup..... 94
Areas..... 96
Networks..... 97
Interfaces..... 98
Virtual Links..... 99
Neighbours..... 99
General Information 100
RIP..... 108
General Information..... 108
General Setup..... 109
Interfaces..... 110
Networks..... 111
Neighbors..... 112
Routes..... 112
General Information 113
Routes, Equal Cost Multipath Routing, Policy Routing..... 116
General Information 116
Routes..... 117
General Information 118
General Interface Settings..... 121
General Information 121
Interface Status..... 121
Traffic Monitoring..... 122
ARLAN 655 Wireless Client Card..... 123
General Information..... 123
Installation..... 123
Wireless Interface Configuration..... 124
Troubleshooting..... 125
Interface Bonding..... 127
General Information 127
General Information 130
Bridge..... 133
General Information..... 134
Bridge Interface Setup..... 135
Port Settings..... 136
Bridge Monitoring..... 137
Bridge Port Monitoring..... 137
Bridge Host Monitoring..... 138
Bridge Firewall General Description..... 138

Bridge Packet Filter....142
Bridge NAT....143
Bridge Brouting Facility....144
Troubleshooting....144
CISCO/Aironet 2.4GHz 11Mbps Wireless Interface....146
General Information....146
Wireless Interface Configuration....147
Troubleshooting....150
Application Examples....150
Cyclades PC300 PCI Adapters....155
General Information....155
Synchronous Interface Configuration....156
Troubleshooting....157
RSV/V.35 Synchronous Link Applications....157
Ethernet Interfaces....160
General Information....160
Ethernet Interface Configuration....161
Monitoring the Interface Status....162
Troubleshooting....162
FarSync X.21 Interface....164
General Information....164
Synchronous Interface Configuration....165
Troubleshooting....166
Synchronous Link Applications....166
FrameRelay (PVC, Private Virtual Circuit) Interface....172
General Information....172
Configuring Frame Relay Interface....173
Frame Relay Configuration....173
Troubleshooting....177
GPRS PCMCIA....178
How to make a GPRS connection....178
ISDN (Integrated Services Digital Network) Interface....180
General Information....180
ISDN Hardware and Software Installation....181
ISDN Client Interface Configuration....182
ISDN Server Interface Configuration....183
ISDN Examples....184
LMC/SBEI Synchronous Interfaces....188
General Information....188
Synchronous Interface Configuration....188
General Information....189
M3P....191
General Information....191
Setup....192
MOXA C101 Synchronous Interface....194
General Information....194

Synchronous Interface Configuration..... 195
Troubleshooting.....197
Synchronous Link Application Examples.....197
MOXA C502 Dual-port Synchronous Interface.....202
General Information..... 202
Synchronous Interface Configuration..... 203
Troubleshooting.....204
Synchronous Link Application Examples.....204
PPP and Asynchronous Interfaces..... 209
General Information..... 209
Serial Port Configuration.....210
PPP Server Setup.....211
PPP Client Setup..... 212
PPP Application Example..... 213
RadioLAN 5.8GHz Wireless Interface.....215
General Information..... 215
Wireless Interface Configuration..... 216
Troubleshooting.....218
Wireless Network Applications.....218
Wireless Client and Wireless Access Point Manual..... 220
General Information..... 222
Wireless Interface Configuration..... 224
Nstreme Settings.....230
Nstreme2 Group Settings..... 231
Registration Table..... 234
Connect List..... 235
Access List..... 236
Info..... 237
Virtual Access Point Interface.....240
WDS Interface Configuration..... 241
Align.....242
Align Monitor.....243
Frequency Monitor.....244
Manual Transmit Power Table.....244
Network Scan..... 245
Security Profiles..... 246
Sniffer.....248
Sniffer Sniff.....249
Sniffer Packets.....250
Snooper.....250
General Information 251
Troubleshooting.....260
Xpeed SDSL Interface.....262
General Information..... 262
Xpeed Interface Configuration.....263
Frame Relay Configuration Examples..... 264
Troubleshooting.....265

EoIP Tunnel Interface..... 267
General Information..... 267
EoIP Setup..... 268
EoIP Application Example..... 269
Troubleshooting..... 271
IP Security..... 272
General Information 272
Policy Settings..... 275
Peers..... 277
Remote Peer Statistics..... 278
Installed SAs..... 279
Flushing Installed SA Table..... 280
Counters..... 281
General Information 282
IPIP Tunnel Interfaces..... 288
General Information..... 288
IPIP Setup..... 289
General Information 290
L2TP Interface..... 292
General Information..... 292
L2TP Client Setup..... 294
Monitoring L2TP Client..... 295
L2TP Server Setup..... 295
L2TP Server Users..... 296
L2TP Application Examples..... 297
Troubleshooting..... 301
PPPoE..... 303
General Information..... 303
PPPoE Client Setup..... 305
Monitoring PPPoE Client..... 306
PPPoE Server Setup (Access Concentrator)..... 307
PPPoE Server Users..... 308
Application Examples..... 309
Troubleshooting..... 310
PPTP..... 312
General Information..... 312
PPTP Client Setup..... 314
Monitoring PPTP Client..... 315
PPTP Server Setup..... 315
PPTP Server Users..... 316
PPTP Application Examples..... 317
Troubleshooting..... 320
VLAN..... 321
General Information..... 321
VLAN Setup..... 323
Application Example..... 323

Graphing..... 325
General Information..... 325
General Options..... 326
Health Graphing..... 326
Interface Graphing..... 327
Simple Queue Graphing..... 327
Resource Graphing..... 328
HotSpot User AAA..... 329
General Information..... 329
HotSpot User Profiles..... 330
HotSpot Users..... 331
HotSpot Active Users..... 333
IP accounting..... 335
General Information..... 335
Local IP Traffic Accounting..... 336
Local IP Traffic Accounting Table..... 337
Web Access to the Local IP Traffic Accounting Table..... 338
PPP User AAA..... 339
General Information..... 339
Local PPP User Profiles..... 340
Local PPP User Database..... 342
Monitoring Active PPP Users..... 343
PPP User Remote AAA..... 344
RADIUS client..... 345
General Information..... 345
RADIUS Client Setup..... 346
Suggested RADIUS Servers..... 347
Supported RADIUS Attributes..... 348
Troubleshooting..... 353
Router User AAA..... 355
General Information..... 355
Router User Groups..... 356
Router Users..... 357
Monitoring Active Router Users..... 358
Router User Remote AAA..... 359
Traffic Flow..... 360
General Information..... 360
General Configuration..... 361
Traffic-Flow Target..... 361
General Information..... 361
Bandwidth Control..... 363
General Information..... 363
Queue Types..... 368
Interface Default Queues..... 371
Simple Queues..... 371
Queue Trees..... 372

General Information 373
Filter 377
General Information 377
Firewall Filter 378
Filter Applications 384
Address Lists 386
General Information 386
Address Lists 386
Mangle 388
General Information 388
Mangle 388
General Information 393
NAT 395
General Information 395
NAT 395
NAT Applications 401
Packet Flow 403
General Information 403
Packet Flow 403
General Firewall Information 405
DHCP Client and Server 408
General Information 409
DHCP Client Setup 410
DHCP Server Setup 412
Store Leases on Disk 414
DHCP Networks 414
DHCP Server Leases 415
DHCP Alert 417
DHCP Option 418
DHCP Relay 419
Question&Answer-Based Setup 420
General Information 421
DNS Client and Cache 424
General Information 424
Client Configuration and Cache Setup 425
Cache Monitoring 426
Static DNS Entries 426
Flushing DNS cache 426
HotSpot Gateway 428
General Information 429
Question&Answer-Based Setup 434
HotSpot Interface Setup 435
HotSpot Server Profiles 436
HotSpot User Profiles 438
HotSpot Users 438
HotSpot Active Users 438

HotSpot Cookies..... 438
HTTP-level Walled Garden.....439
IP-level Walled Garden..... 440
One-to-one NAT static address bindings..... 441
Active Host List.....441
Service Port..... 442
Customizing HotSpot: Firewall Section.....443
Customizing HotSpot: HTTP Servlet Pages..... 445
Possible Error Messages.....451
HotSpot How-to's..... 453
HTTP Proxy..... 454
General Information 454
Setup.....455
Access List..... 456
Direct Access List..... 457
HTTP Methods.....458
IP Pools..... 460
General Information 460
Setup.....461
Used Addresses from Pool..... 461
SOCKS Proxy Server..... 463
General Information 463
SOCKS Configuration.....464
Access List..... 465
Active Connections..... 465
General Information 466
UPnP..... 468
General Information 468
Enabling Universal Plug-n-Play.....469
UPnP Interfaces.....469
Web Proxy..... 472
General Information 472
Setup.....474
Access List..... 475
Direct Access List..... 477
Cache Management.....478
Complementary Tools..... 478
Transparent Mode.....479
HTTP Methods.....479
Certificate Management..... 482
General Information 482
Certificates.....483
DDNS Update Tool..... 486
General Information 486
Dynamic DNS Update.....487
GPS Synchronization..... 488

General Information 488
Synchronizing with a GPS Receiver 489
GPS Monitoring 490
LCD Management 491
General Information 491
Configuring the LCD's Settings 493
LCD Information Display Configuration 494
LCD Troubleshooting 495
MNDP 496
General Information 496
Setup 497
Neighbour List 497
NTP (Network Time Protocol) 499
General Information 499
Client 500
Server 501
Time Zone 501
RouterBoard-specific functions 503
General Information 503
BIOS upgrading 504
BIOS Configuration 505
System Health Monitoring 506
LED Management or RB200 507
LED Management on RB500 508
Fan voltage control 508
Console Reset Jumper 509
Support Output File 510
General Information 510
Generating Support Output File 510
System Resource Management 511
General Information 512
System Resource 512
IRQ Usage Monitor 513
IO Port Usage Monitor 513
USB Port Information 514
PCI Information 514
Reboot 515
Shutdown 515
Router Identity 516
Date and Time 516
Configuration Change History 517
System Note 518
Bandwidth Test 519
General Information 519
Server Configuration 520
Client Configuration 521

ICMP Bandwidth Test 523
General Information 523
ICMP Bandwidth Test 523
Packet Sniffer 525
General Information 525
Packet Sniffer Configuration 526
Running Packet Sniffer 527
Sniffed Packets 528
Packet Sniffer Protocols 529
Packet Sniffer Host 531
Packet Sniffer Connections 531
Ping 533
General Information 533
The Ping Command 534
MAC Ping Server 535
Torch (Realtime Traffic Monitor) 536
General Information 536
The Torch Command 536
Traceroute 539
General Information 539
The Traceroute Command 540
Scheduler 541
General Information 541
Network Watching Tool 541
Serial Port Monitor 544
General Information 544
Sigwatch 544
Scripting Host 547
General Information 548
Console Command Syntax 548
Expression Grouping 550
Variables 551
Command Substitution and Return Values 551
Operators 552
Data types 555
Command Reference 556
Special Commands 561
Additional Features 562
Script Repository 562
Task Management 563
Script Editor 564
Scheduler 566
General Information 566
Scheduler Configuration 566
Traffic Monitor 569
General Information 569

Traffic Monitor....569
IP Telephony....571
General Information....572
General Voice port settings....574
Voicetronix Voice Ports....575
LineJack Voice Ports....576
PhoneJack Voice Ports....578
Zaptel Voice Ports....580
ISDN Voice Ports....581
Voice Port for Voice over IP (voip)....583
Numbers....583
Regional Settings....586
Audio CODECs....587
AAA....587
Gatekeeper....589
Troubleshooting....592
A simple example....592
System Watchdog....600
General Information....600
Hardware Watchdog Management....600
UPS Monitor....602
General Information....602
UPS Monitor Setup....603
Runtime Calibration....604
UPS Monitoring....605
VRRP....607
General Information....607
VRRP Routers....608
Virtual IP addresses....609
A simple example of VRRP fail over....610

Device Driver List

Document revision 3 (Thu Jul 14 18:27:43 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
Ethernet
  Specifications
  Description
  Notes
Wireless
  Specifications
  Description
Aironet Arlan
  Specifications
  Description
RadioLAN
  Specifications
  Description
Synchronous Serial
  Specifications
  Description
Asynchronous Serial
  Specifications
  Description
ISDN
  Specifications
  Description
VoIP
  Specifications
  Description
xDSL
  Specifications
  Description
HomePNA
  Specifications
  Description
LCD
  Specifications
  Description
PCMCIA Adapters
  Specifications
  Description
GPRS Cards
  Specifications

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

General Information

Summary

The document lists the drivers, included in MikroTik RouterOS and the devices that are tested to work with MikroTik RouterOS. If a device is not listed here, it does not mean the device is not supported, it still may work. It just means that the device was not tested.

Ethernet

Packages required: ***system***

Description

3Com 509 Series

Chipset type: 3Com 509 Series ISA 10Base

Compatibility:

- 3Com EtherLink III

3Com FastEtherLink

Chipset type: 3Com 3c590/3c900 (3Com FastEtherLink and FastEtherLink XL) PCI 10/100Base

Compatibility:

- 3c590 Vortex 10BaseT
- 3c592 chip
- 3c595 Vortex 100BaseTX
- 3c595 Vortex 100BaseT4
- 3c595 Vortex 100Base-MII
- 3c597 chip
- 3Com Vortex
- 3c900 Boomerang 10BaseT
- 3c900 Boomerang 10Mbit/s Combo
- 3c900 Cyclone 10Mbit/s Combo
- 3c900B-FL Cyclone 10Base-FL
- 3c905 Boomerang 100BaseTX
- 3c905 Boomerang 100BaseT4
- 3c905B Cyclone 100BaseTX

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- 3c905B Cyclone 10/100/BNC
- 3c905B-FX Cyclone 100BaseFX
- 3c905C Tornado
- 3c980 Cyclone
- 3cSOHO100-TX Hurricane
- 3CSOHO100B-TX
- 3c555 Laptop Hurricane
- 3c575 Boomerang CardBus
- 3CCFE575 Cyclone CardBus
- 3CCFE656 Cyclone CardBus
- 3c575 series CardBus
- 3Com Boomerang

ADMtek Pegasus

Chipset type: ADMtek Pegasus/Pegasus II USB 10/100BaseT

Compatibility:

- Planet 10/100Base-TX USB Ethernet Adapter UE-9500
- Linksys Instant EtherFast 10/100 USB Network Adapter USB100TX

AMD PCnet

Chipset type: AMD PCnet/PCnet II ISA/PCI 10BaseT

Compatibility:

- AMD PCnet-ISA
- AMD PCnet-ISA II
- AMD PCnet-PCI II
- AMD 79C960 based cards

AMD PCnet32

Chipset type: AMD PCnet32 PCI 10BaseT and 10/100BaseT

Compatibility:

- AMD PCnet-PCI
- AMD PCnet-32
- AMD PCnet-Fast

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Broadcom Tigon3

Chipset type: Broadcom Tigon3 PCI 10/100/1000BaseT

Compatibility:

- Broadcom Tigon3 570x
- Broadcom Tigon3 5782
- Broadcom Tigon3 5788
- Broadcom Tigon3 5901
- Broadcom Tigon3 5901-2
- SysKonnect SK-9Dxx Gigabit Ethernet
- SysKonnect SK-9Mxx Gigabit Ethernet
- Altima AC100x
- Altima AC9100

Davicom DM9102

Chipset type: Davicom DM9102 PCI 10/100Base

Compatibility:

- Davicom DM9102
- Davicom DM9102A
- Davicom DM9102A+DM9801
- Davicom DM9102A+DM9802

DEC 21x4x 'Tulip'

Chipset type: DEC 21x4x "Tulip" PCI 10/100Base

Compatibility:

- Digital DC21040 Tulip
- Digital DC21041 Tulip
- Digital DS21140 Tulip
- 21140A chip
- 21142 chip
- Digital DS21143 Tulip
- D-Link DFE 570TX 4-port
- Lite-On 82c168 PNIC
- Macronix 98713 PMAC

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- Macronix 98715 PMAC
- Macronix 98725 PMAC
- ASIX AX88140
- Lite-On LC82C115 PNIC-II
- ADMtek AN981 Comet
- Compex RL100-TX
- Intel 21145 Tulip
- IMC QuikNic FX
- Conexant LANfinity

Intel EtherExpressPro

Chipset type: Intel i82557 "Speedo3" (Intel EtherExpressPro) PCI 10/100Base

Compatibility:

- Intel i82557/i82558/i82559ER/i82801BA-7 EtherExpressPro PCI cards

Intel PRO/1000

Chipset type: Intel i8254x (Intel PRO/1000) PCI 10/100/1000Base

Compatibility:

- Intel PRO/1000 Gigabit Server Adapter (i82542, Board IDs: 700262-xxx, 717037-xxx)
- Intel PRO/1000 F Server Adapter (i82543, Board IDs: 738640-xxx, A38888-xxx)
- Intel PRO/1000 T Server Adapter (i82543, Board IDs: A19845-xxx, A33948-xxx)
- Intel PRO/1000 XT Server Adapter (i82544, Board IDs: A51580-xxx)
- Intel PRO/1000 XF Server Adapter (i82544, Board IDs: A50484-xxx)
- Intel PRO/1000 T Desktop Adapter (i82544, Board IDs: A62947-xxx)
- Intel PRO/1000 MT Desktop Adapter (i82540, Board IDs: A78408-xxx, C91016-xxx)
- Intel PRO/1000 MT Server Adapter (i82545, Board IDs: A92165-xxx, C31527-xxx)
- Intel PRO/1000 MT Dual Port Server Adapter (i82546, Board IDs: A92111-xxx, C29887-xxx)
- Intel PRO/1000 MT Quad Port Server Adapter (i82546, Board IDs: C32199-xxx)
- Intel PRO/1000 MF Server Adapter (i82545, Board IDs: A91622-xxx, C33915-xxx)
- Intel PRO/1000 MF Server Adapter (LX) (i82545, Board IDs: A91624-xxx, C33916-xxx)
- Intel PRO/1000 MF Dual Port Server Adapter (i82546, Board IDs: A91620-xxx, C30848-xxx)
- Intel PRO/1000 GT Desktop Adapter (i82541PI)

Marvell Yukon

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Chipset type: Marvell Yukon 88E80xx PCI 10/100/1000Base

Compatibility:

- 3Com 3C940 Gigabit LOM Ethernet Adapter
- 3Com 3C941 Gigabit LOM Ethernet Adapter
- Allied Telesyn AT-2970LX Gigabit Ethernet Adapter
- Allied Telesyn AT-2970LX/2SC Gigabit Ethernet Adapter
- Allied Telesyn AT-2970SX Gigabit Ethernet Adapter
- Allied Telesyn AT-2970SX/2SC Gigabit Ethernet Adapter
- Allied Telesyn AT-2970TX Gigabit Ethernet Adapter
- Allied Telesyn AT-2970TX/2TX Gigabit Ethernet Adapter
- Allied Telesyn AT-2971SX Gigabit Ethernet Adapter
- Allied Telesyn AT-2971T Gigabit Ethernet Adapter
- DGE-530T Gigabit Ethernet Adapter
- EG1032 v2 Instant Gigabit Network Adapter
- EG1064 v2 Instant Gigabit Network Adapter
- Marvell 88E8001 Gigabit LOM Ethernet Adapter
- Marvell RDK-80xx Adapter
- Marvell Yukon Gigabit Ethernet 10/100/1000Base-T Adapter
- N-Way PCI-Bus Giga-Card 1000/100/10Mbps(L)
- SK-9521 10/100/1000Base-T Adapter
- SK-98xx Gigabit Ethernet Server Adapter
- SMC EZ Card 1000
- Marvell Yukon 88E8010 based
- Marvell Yukon 88E8003 based
- Marvell Yukon 88E8001 based

National Semiconductor DP83810

Chipset type: National Semiconductor DP83810 PCI 10/100BaseT

Compatibility:

- RouterBoard 200 built-in Ethernet
- RouterBoard 24 4-port Ethernet
- NS DP8381x-based cards

National Semiconductor DP83820

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Chipset type: National Semiconductor DP83820 PCI 10/100/1000BaseT

Compatibility:

- Planet ENW-9601T
- NS DP8382x-based cards

NE2000 ISA

Chipset type: NE2000 ISA 10Base

Compatibility:

- various ISA cards

NE2000 PCI

Chipset type: NE2000 PCI 10Base

Compatibility:

- RealTek RTL-8029
- Winbond 89C940 and 89C940F
- Compex RL2000
- KTI ET32P2
- NetVin NV5000SC
- Via 86C926
- SureCom NE34
- Holtek HT80232
- Holtek HT80229
- IMC EtherNic/PCI FO

NS8390

Chipset type: NS8390 PCMCIA/CardBus 10Base

Compatibility:

- D-Link DE-660 Ethernet
- NE-2000 Compatible PCMCIA Ethernet
- NS8390-based PCMCIA cards

RealTek RTL8129

Chipset type: RealTek RTL8129 PCI 10/100Base

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Compatibility:

- RealTek RTL8129 Fast Ethernet
- RealTek RTL8139 Fast Ethernet
- RTL8139A/B/C/D chip
- RTL8130 chip
- RTL8100B chip
- SMC1211TX EZCard 10/100 (RealTek RTL8139)
- Accton MPX5030 (RealTek RTL8139)
- D-Link DFE 538TX

RealTek RTL8169

Chipset type: RealTek RTL8169 PCI 10/100/1000Base

Compatibility:

- RealTek RTL8169 Gigabit Ethernet (**not** recommended: may lock up the router)

Sundance ST201 'Alta'

Chipset type: Sundance ST201 "Alta" PCI 10/100Base

Compatibility:

- D-Link DFE-550TX Fast Ethernet Adapter
- D-Link DFE-550FX 100Mbps Fiber-optics Adapter
- D-Link DFE-580TX 4-port Server Adapter (**not** recommended: may lock up the system)
- D-Link DFE-530TXS Fast Ethernet Adapter
- D-Link DL10050-based FAST Ethernet Adapter
- Sundance ST201 "Alta" chip
- Kendin KS8723 chip

TI ThunderLAN

Chipset type: TI ThunderLAN PCI 10/100Base

Compatibility:

- Compaq Netelligent 10 T
- Compaq Netelligent 10 T/2
- Compaq Netelligent 10/100 TX
- Compaq NetFlex-3/P
- Olicom OC-2183

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- Olicom OC-2185
- Olicom OC-2325
- Olicom OC-2326

VIA vt612x 'Velocity'

Chipset type: VIA vt612x "Velocity" PCI 10/100/1000Base

Compatibility:

- VIA VT6120
- VIA VT6121
- VIA VT6122

VIA vt86c100 'Rhine'

Chipset type: VIA vt86c100 "Rhine" PCI 10/100Base

Compatibility:

- VIA Rhine (vt3043)
- VIA Rhine II (vt3065 AKA vt86c100)
- VIA VT86C100A Rhine
- VIA VT6102 Rhine-II
- VIA VT6105 Rhine-III
- VIA VT6105M Rhine-III
- RouterBOARD 44 4-port Fast Ethernet card
- D-Link DFE 530TX

Winbond w89c840

Chipset type: Winbond w89c840 PCI 10/100Base

Compatibility:

- Winbond W89c840
- Compex RL100-ATX

Notes

For ISA cards load the driver by specifying the I/O base address. IRQ is not required.

Wireless

Packages required: ***wireless***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

Atheros

Chipset type: Atheros AR5001X PC/PCI 11/54Mbit/s IEEE802.11a/b/g

Compatibility:

- Intel 5000 series
- Dlink DWL-A520
- Dlink DWL-G650
- Atheros AR5000 chipset series based IEEE802.11a (AR5210 MAC plus AR5110 PHY chips) cards
- Atheros AR5001A chipset series based IEEE802.11a (AR5211 MAC plus AR5111 PHY chips) cards
- Atheros AR5001X chipset series based IEEE802.11a (AR5211 MAC plus AR5111 PHY chips), IEEE802.11b/g (AR5211 MAC plus AR2111 PHY chips), IEEE802.11a/b/g (AR5211 MAC plus AR5111 and 2111 PHY chips) cards
- Atheros AR5001X+ chipset series based IEEE802.11a (AR5212 MAC plus AR5111 PHY chips), IEEE802.11b/g (AR5212 MAC plus AR2111 PHY chips), IEEE802.11a/b/g (AR5212 MAC plus AR5111 and 2111 PHY chips) cards
- Atheros AR5002X+ chipset series based IEEE802.11b/g (AR5212 MAC plus AR2112 PHY chips), IEEE802.11a/b/g (AR5212 MAC plus AR5112 PHY chips) cards
- Atheros AR5004X+ chipset series based IEEE802.11b/g (AR5213 MAC plus AR2112 PHY chips), IEEE802.11a/b/g (AR5213 MAC plus AR5112 PHY chips) cards
- Atheros AR5006X chipset series based IEEE802.11a/b/g (AR5413/AR5414 single-chip devices) cards

Cisco/Aironet

Chipset type: Cisco/Aironet ISA/PCI/PC 11Mbit/s IEEE802.11b

Compatibility:

- Aironet ISA/PCI/PC4800 2.4GHz DS 11Mbit/s Wireless LAN Adapters (100mW)
- Aironet ISA/PCI/PC4500 2.4GHz DS 2Mbit/s Wireless LAN Adapters (100mW)
- CISCO AIR-PCI340 2.4GHz DS 11Mbit/s Wireless LAN Adapters (30mW)
- CISCO AIR-PCI/PC350/352 2.4GHz DS 11Mbit/s Wireless LAN Adapters (100mW)

Intersil Prism II

Chipset type: Intersil Prism II PC/PCI 11Mbit/s IEEE802.11b

Compatibility:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- Intersil PRISM2 Reference Design 11Mbit/s IEEE802.11b WLAN Card
- GemTek WL-211 Wireless LAN PC Card
- Compaq iPaq HNW-100 11Mbit/s 802.11b WLAN Card
- Samsung SWL2000-N 11Mbit/s 802.11b WLAN Card
- Z-Com XI300 11Mbit/s 802.11b WLAN Card
- ZoomAir 4100 11Mbit/s 802.11b WLAN Card
- Linksys WPC11 11Mbit/s 802.11b WLAN Card
- Addtron AWP-100 11Mbit/s 802.11b WLAN Card
- D-Link DWL-650 11Mbit/s 802.11b WLAN Card
- SMC 2632W 11Mbit/s 802.11b WLAN Card
- BroMax Freeport 11Mbit/s 802.11b WLAN Card
- Intersil PRISM2 Reference Design 11Mbit/s WLAN Card
- Bromax OEM 11Mbit/s 802.11b WLAN Card (Prism 2.5)
- corega K.K. Wireless LAN PCC-11
- corega K.K. Wireless LAN PCCA-11
- CONTEC FLEXSCAN/FX-DDS110-PCC
- PLANEX GeoWave/GW-NS110
- Ambicom WL1100 11Mbit/s 802.11b WLAN Card
- LeArtery SYNCBYAIR 11Mbit/s 802.11b WLAN Card
- Intermec MobileLAN 11Mbit/s 802.11b WLAN Card
- NETGEAR MA401 11Mbit/s 802.11 WLAN Card
- Intersil PRISM Freedom 11Mbit/s 802.11 WLAN Card
- OTC Wireless AirEZY 2411-PCC 11Mbit/s 802.11 WLAN Card
- Z-Com XI-325HP PCMCIA 200mW Card
- Z-Com XI-626 Wireless PCI Card

Note If planned to use WEP with Prism cards see link for more information: ***Wireless Security***

WaveLAN/ORiNOCO

Chipset type: Lucent/Agere/Proxim WaveLAN/ORiNOCO ISA/PC 11Mbit/s IEEE802.11b

Compatibility:

- WaveLAN Bronze/Gold/Silver ISA/PCMCIA

Aironet Arlan

Packages required: ***arlan***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

This is driver for legacy Aironet Arlan cards, not for newer Cisco/Aironet cards.

Chipset type: Aironet Arlan IC2200 ISA 2Mbit/s 2.4GHz

Compatibility:

- Aironet Arlan 655

RadioLAN

Packages required: ***radiolan***

Description

This is driver for legacy RadioLAN cards.

Chipset type: RadioLAN ISA/PC 10Mbit/s 5.8GHz

Compatibility:

- RadioLAN ISA card (Model 101)
- RadioLAN PCMCIA card

Synchronous Serial

Packages required: ***synchronous***

Description

- Moxa C101 ISA and PCI V.35 (4 Mbit/s)
- Moxa C502 PCI 2-port V.35 (8 Mbit/s)
- Cyclades PCI PC-300 V.35 (5 Mbit/s)
- Cyclades PCI PC-300 E1/T1
- FarSync PCI V.35/X.21 (8.448 Mbit/s)
- LMC/SBEI wanPCI-1T1E1 PCI T1/E1 (also known as DS1 or LMC1200P, 1.544 Mbit/s or 2.048 Mbit/s)
- LMC/SBEI wanPCI-1T3 PCI T3 (also known as DS3, 44.736Mbit/s)
- Sangoma S5141 (dual-port) and S5142 (quad-port) PCI RS232/V.35/X.21 (4Mbit/s - primary port and 512Kbit/s - secondary ones)
- Sangoma S5148 (single-port) and S5147 (dual-port) PCI E1/T1

Asynchronous Serial

Packages required: ***system***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

- Standard Communication Ports Com1 and Com2
- Moxa Smartio C104H/PCI, CP-114, CT-114, CP-132, C168H, CP-168H, and CP-168U PCI 2/4/8 port up to 4 cards (up to 32 ports)
- Cyclades Cyclom-Y and Cyclades-Z Series PCI cards up to 64 ports per card, up to 4 cards (up to 256 ports)
- TCL DataBooster 4 or 8 PCI 4/8 port cards
- Sangoma S514/56 PCI 56 or 64Kbit/s DDS DSU with secondary 128Kbit/s RS232 port (**Note**: this card is not for modem pools or serial terminals)

ISDN

Packages required: ***isdn***

Description

PCI ISDN cards:

- Eicon.Diehl Diva PCI
- Sedlbauer Speed Card PCI
- ELSA Quickstep 1000PCI
- Traverse Technologie NETjet PCI S0 card
- Teles PCI
- Dr. Neuhaus Niccy PCI
- AVM Fritz PCI
- Gazel PCI ISDN cards
- HFC-2BS0 based PCI cards (TeleInt SA1)
- Winbond W6692 based PCI cards

VoIP

Packages required: ***telephony***

Description

H.323 Protocol VoIP Analog Gateways

- QuickNet LineJack ISA
- QuickNet PhoneJack ISA
- Voicetronix V4PCI - 4 analog telephone lines cards
- Zaptel X.100P IP telephony card (1 analog line)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

xDSL

Packages required: ***synchronous***

Description

Xpeed 300 SDSL cards (up to 6.7km twisted pair wire connection, max 2.3Mbit/s)

HomePNA

Packages required: ***system***

Description

Linksys HomeLink PhoneLine Network Card (up to 10Mbit/s home network over telephone line)

LCD

Packages required: ***lcd***

Description

- Crystalfontz Intelligent Serial LCD Module 632 (16x2 characters) and 634 (20x4 characters)
- Powertip Character LCD Module PC1602 (16x2 characters), PC1604 (16x4 characters), PC2002 (20x2 characters), PC2004 (20x4 characters), PC2402 (24x2 characters) and PC2404 (24x4 characters)

PCMCIA Adapters

Packages required: ***system***

Description

- Vadem VG-469 PCMCIA-ISA adapter (one or two PCMCIA ports)
- RICOH PCMCIA-PCI Bridge with R5C475 II or RC476 II chip (one or two PCMCIA ports)
- CISCO/Aironet PCMCIA adapter (ISA and PCI versions) for CISCO/Aironet PCMCIA cards only

GPRS Cards

Packages required: ***wireless***

Description

- NWH 1600 GPRS Modem (Benq M32 chip)
  For more information, see ***interface list***.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

License Management

Document revision 3.1 (Thu Mar 03 11:06:06 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Description
License Management
  Description
  Property Description
  Command Description

General Information

Summary

MikroTik RouterOS software has a licensing system with Software License (Software Key) issued for each individual installation of the RouterOS.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/system license***
Hardware usage: ***Not significant***

Description

The Software License can be obtained through the Account Server at **www.mikrotik.com** after the MikroTik RouterOS has been installed. The Software ID of the installation is required when obtaining the Software License. Please read the MikroTik RouterOS Basic Setup Guide for detailed explanation of the installation and licensing process.

RouterOS allows you to use all its features without registration for about 24 hours from the first run. Note that if you shut the router down, the countdown is paused, and it is resumed only when the router is started again. During this period you must get a key, otherwise you will need to reinstall the system. A purchased license key allows you to use RouterOS features according to the chosen license level for unlimited time, and gives you rights to freely upgrade and downgrade its versions for the term of one year since the key was purchased. A free registred license key (referred as a SOHO key further on) allows you to use a restricted set of functions for unlimited period of time, but does not allows upgrading and downgrading versions.

There are 6 licensing levels, each providing some additional features. Level 0 means that there is no key and all the features are enabled for one day. Level 2 is a transitional license level from versions

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

prior 2.8, that allows to use all the features were allowed by your original license key for a previus version.

Level number	1 (SOHO)	3 (ISP)	4 (WISP)	5 (WISP AP)	6 (Controller)
Wireless Client and Bridge	-	-	yes	yes	yes
Wireless AP	-	-	-	yes	yes
Synchronous interfaces	-	-	yes	yes	yes
EoIP tunnels	1	unlimited	unlimited	unlimited	unlimited
PPPoE tunnels	1	200	200	500	unlimited
PPTP tunnels	1	200	200	unlimited	unlimited
L2TP tunnels	1	200	200	unlimited	unlimited
VLAN interfaces	1	unlimited	unlimited	unlimited	unlimited
P2P firewall rules	1	unlimited	unlimited	unlimited	unlimited
NAT rules	1	unlimited	unlimited	unlimited	unlimited
HotSpot active users	1	1	200	500	unlimited
RADIUS client	-	yes	yes	yes	yes
Queues	1	unlimited	unlimited	unlimited	unlimited
Web proxy	-	yes	yes	yes	yes
RIP, OSPF, BGP protocols	-	yes	yes	yes	yes
Upgrade	configuration erased on upgrade	yes	yes	yes	yes

Note that **Wireless Client and Bridge** means that wireless cards can be used in **station** and **bridge** modes. **Bridge** mode allows one wireless station to connect it.

There is a possibility to upgrade your key (i.e. to extend licensing term) from the console or WinBox.

Note that the license is kept on hard drive. You can move the hard drive to another system, but you can not move license on another hard drive. License transfer to another drive is a paid service (unless your hard drive has crashed). Please contact support@mikrotik.com to arrange this. Also note that you must not use MS-DOS format or fdisk utilities or you may loose the license.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Important: the abovementioned limits depict the limits enforced by the license. The actual number of concurrent tunnels, rules, queues, users, etc. will vary depending the combination of features used and the load they place on the MikroTik RouterOS.

License Management

Home menu level: ***/system license***

Description

There are three methods of entering a key to the system console:

- import a file that should be sent to you after you will require a key (you should upload this file to the router's FTP server)
- simply copy the received key as a text and paste (or type) in to the router's console (no matter in which submenu)

These methods also apply to WinBox, with the difference that key importing and exporting is happening through the Windows host PC itself. The options available:

- **Paste Key** - get a new license from the Windows Clipboard
- **Import Key** - get a new license from a file stored locally on the Windows PC
- **Export Key** - save the existing license as a file on the Windows PC
- **Upgrade/Get New Key** - the same as new-upgrade-key command in system console
- **Update Key** - the same as update-key command in system console

Property Description

key (*read-only: text*) - software license key that unlocks the installation

level (*read-only: integer*: 0..6) - license level of the installation

software-id (*read-only: text*) - ID number of the installation

upgradable-until (*read-only: text*) - the date until which the software version can be upgraded or downgraded

Command Description

import - import a key file (*name*) - file name to use as a key

new-upgrade-key - request a new key (*IP address*) - key server's IP address (*text*) - username to log into the key server (*text*) - password to log into the key server (*integer*: 2..6) - license level to request (*credit-card* | *credit-keys* | *credit-money* | *debit-keys* | *debit-money*) - Payment method to use (*text*; default: "") - script to execute while the command is running (*time*; default: **1s**) - how frequently to execute the given script - if specified, executes the sctipt once, and then terminates the command - command's execution status

- **Resolving www.mikrotik.com** - resolving DNS name
- **Failed to resolve www.mikrotik.com, check your dns settings** - check whether DNS client is set up on the router, and that it is allowed to resolve a DNS name on the DNS server set
- **Failed to connect, probably no IP address** - self-explanatory

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **Failed to connect, is your router public?** - check whether the router has a default route and is able to reack the key server
- **Connecion failed** - connection has timed out
- **Bad response from server** - try again
- **ERROR: You don't have appropriate debit key!** - no existing debit keys on your account matches the requested one
- **ERROR: You don't have enought debit money!** - self-explanatory
- **ERROR: Credit key limit exceeded!** - self-explanatory
- **ERROR: Your credit limit is exceeded!** - self-explanatory
- **ERROR: This payment method is not more allowed! Go to www.mikrotik.com, log on and purchase key there or use other payment methods.** - you can not use the selected payment method from the router anymore due to system changes (for credit cards now)
- **ERROR: You must enable this feature in account server (change user information section)!** - you should enable Allow to use my account in netinstall feature on the accout server (in change user information section
- **ERROR: Incorrect username or password!** - self-explanatory
- **ERROR: You are not allowed to use this service!** - please contact sales@mikrotik.com for further assistance
- **Key upgraded successfully** - the upgrade procedure has been completed successfully

output - exports the current key to a key file

update-key - request a free update of your existing key to the version's 2.9 one (this can be done during your existing key upgrade term) (*IP address*) - key server's IP address (*text*) - username to log into the key server (*text*) - password to log into the key server (*text*; default: **""**) - script to execute while the command is running (*time*; default: **1s**) - how frequently to execute the given script - if specified, executes the sctipt once, and then terminates the command - command's execution status

- **Resolving www.mikrotik.com** - resolving DNS name
- **Failed to resolve www.mikrotik.com, check your dns settings** - check whether DNS client is set up on the router, and that it is allowed to resolve a DNS name on the DNS server set
- **Failed to connect, probably no IP address** - self-explanatory
- **Failed to connect, is your router public?** - check whether the router has a default route and is able to reack the key server
- **Connecion failed** - connection has timed out
- **Bad response from server** - try again
- **ERROR: You must enable this feature in account server (change user information section)!** - you should enable Allow to use my account in netinstall feature on the accout server (in change user information section
- **ERROR: Incorrect username or password!** - self-explanatory
- **ERROR: Someone has already converted this key!** - the requested software ID has already been converted to 2.9 version
- **ERROR: Key for specified software ID is expired. You can purchase new key at www.mikrotik.com website!** - you may not update an expired key to the version 2.9, you must purchase a new one

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **ERROR: You are not allowed to use this service!** - please contact sales@mikrotik.com for further assistance
- **Key upgraded successfully** - the upgrade procedure has been completed successfully

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Specifications Sheet

Document revision 2.6 (Mon Mar 14 12:38:07 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
Description

General Information

Description

Major features

- **Firewall and NAT** - stateful packet filtering; Peer-to-Peer protocol filtering; source and destination NAT; classification by source MAC, IP addresses (networks or a list of networks) and address types, port range, IP protocols, protocol options (ICMP type, TCP flags and MSS), interfaces, internal packet and connection marks, ToS (DSCP) byte, content, matching sequence/frequency, packet size, time and more...
- **Routing** - Static routing; Equal cost multi-path routing; Policy based routing (classification done in firewall); RIP v1 / v2, OSPF v2, BGP v4
- **Data Rate Management** - Hierarchical HTB QoS system with bursts; per IP / protocol / subnet / port / firewall mark; PCQ, RED, SFQ, FIFO queue; CIR, MIR, contention ratios, dynamic client rate equalizing (PCQ), bursts, Peer-to-Peer protocol limitation
- **HotSpot** - HotSpot Gateway with RADIUS authentication and accounting; true Plug-and-Play access for network users; data rate limitation; differentiated firewall; traffic quota; real-time status information; walled-garden; customized HTML login pages; iPass support; SSL secure authentication; advertisement support
- **Point-to-Point tunneling protocols** - PPTP, PPPoE and L2TP Access Concentrators and clients; PAP, CHAP, MSCHAPv1 and MSCHAPv2 authentication protocols; RADIUS authentication and accounting; MPPE encryption; compression for PPPoE; data rate limitation; differentiated firewall; PPPoE dial on demand
- **Simple tunnels** - IPIP tunnels, EoIP (Ethernet over IP)
- **IPsec** - IP security AH and ESP protocols; MODP Diffie-Hellman groups 1,2,5; MD5 and SHA1 hashing algorithms; DES, 3DES, AES-128, AES-192, AES-256 encryption algorithms; Perfect Forwarding Secrecy (PFS) MODP groups 1,2,5
- **Proxy** - FTP and HTTP caching proxy server; HTTPS proxy; transparent DNS and HTTP proxying; SOCKS protocol support; DNS static entries; support for caching on a separate drive; access control lists; caching lists; parent proxy support
- **DHCP** - DHCP server per interface; DHCP relay; DHCP client; multiple DHCP networks; static and dynamic DHCP leases; RADIUS support
- **VRRP** - VRRP protocol for high availability
- **UPnP** - Universal Plug-and-Play support

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **NTP** - Network Time Protocol server and client; synchronization with GPS system
- **Monitoring/Accounting** - IP traffic accounting, firewall actions logging, statistics graphs accessible via HTTP
- **SNMP** - read-only access
- **M3P** - MikroTik Packet Packer Protocol for Wireless links and Ethernet
- **MNDP** - MikroTik Neighbor Discovery Protocol; also supports Cisco Discovery Protocol (CDP)
- **Tools** - ping; traceroute; bandwidth test; ping flood; telnet; SSH; packet sniffer; Dynamic DNS update tool

TCP/IP protocol suite:

- **Wireless** - IEEE802.11a/b/g wireless client and Access Point; Nsetreme and Nstreme2 proprietary protocols; Wireless Distribution System (WDS) support; virtual AP; 40 and 104 bit WEP; WAP pre-shared key authentication; access control list; authentication on RADIUS server; roaming (for wireless client); Access Point bridging
- **Bridge** - spanning tree protocol; multiple bridge interfaces; bridge firewalling, MAC NATting
- **VLAN** - IEEE802.1q Virtual LAN support on Ethernet and wireless links; multiple VLANs; VLAN bridging
- **Synchronous** - V.35, V.24, E1/T1, X.21, DS3 (T3) media types; sync-PPP, Cisco HDLC, Frame Relay line protocols; ANSI-617d (ANDI or annex D) and Q933a (CCITT or annex A) Frame Relay LMI types
- **Asynchronous** - serial PPP dial-in / dial-out; PAP, CHAP, MSCHAPv1 and MSCHAPv2 authentication protocols; RADIUS authentication and accounting; onboard serial ports; modem pool with up to 128 ports; dial on demand
- **ISDN** - ISDN dial-in / dial-out; PAP, CHAP, MSCHAPv1 and MSCHAPv2 authentication protocols; RADIUS authentication and accounting; 128K bundle support; Cisco HDLC, x75i, x75ui, x75bui line protocols; dial on demand
- **SDSL** - Single-line DSL support; line termination and network termination modes

Layer 2 connectivity

IA32 Hardware requirements

- **CPU and motherboard** - advanced 4th generation (core frequency 100MHz or more), 5th generation (Intel Pentium, Cyrix 6X86, AMD K5 or comparable) or newer uniprocessor Intel IA-32 (i386) compatible (multiple processors are not supported)
- **RAM** - minimum 32 MiB, maximum 1 GiB; 64 MiB or more recommended
- **Non-volatile storage medium** - standard ATA/IDE interface controller and drive (SCSI and USB controllers and drives are not supported; RAID controllers that require additional drivers are not supported) with minimum of 64 Mb space; Flash and Microdrive devices may be connected using an adapted with ATA interface

MIPS Hardware requiremetns

- **Supported systems** - RouterBOARD 500 series (532, 512 and 511)
- **RAM** - minimum 32 MiB

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **Non-volatile storage medium** - onboard NAND device, 64Mb

Hardware needed for installation time only

- **Floppy-based installation** - standard AT floppy controller and 3.5" disk drive connected as the first floppy disk drive (A); AT, PS/2 or USB keyboard; VGA-compatible video controller card and monitor
- **CD-based installation** - standard ATA/ATAPI interface controller and CD drive supporting "El Torito" bootable CDs (you might need also to check if the router's BIOS supports booting from this type of media; if El Torito is not supported by the BIOS, you can still boot up from the CD using Smart Boot Manager Floppy); AT, PS/2 or USB keyboard; VGA-compatible video controller card and monitor
- **Floppy-based network installation** - standard AT floppy controller and 3.5" disk drive connected as the first floppy disk drive (A); PCI Ethernet network interface card supported by MikroTik RouterOS (see the Device Driver List for the list)
- **Full network-based installation** - PCI Ethernet network interface card supported by MikroTik RouterOS (see the Device Driver List for the list) with PXE or EtherBoot extension booting ROM (you might need also to check if the router's BIOS supports booting from network)

Depending on installation method chosen the router must have the following hardware:

Configuration possibilities

RouterOS provides powerful command-line configuration interface. You can also manage the router through WinBox - the easy-to-use remote configuration GUI for Windows -, which provides all the benefits of the command-line interface, without the actual "command-line", which may scare novice users. Web-based configuration is provided for some most popular functionality. Major features:

- Clean and consistent user interface
- Runtime configuration and monitoring
- Multiple connections
- User policies
- Action history, undo/redo actions
- safe mode operation
- Scripts can be scheduled for executing at certain times, periodically, or on events. All command-line commands are supported in scripts

- **Local teminal console** - AT, PS/2 or USB keyboard and VGA-compatible video controller card with monitor
- **Serial console** - any (you may choose any one; the first, also known as COM1, is used by default) RS232 asynchronous serial port, which is by default set to 9600bit/s, 8 data bits, 1 stop bit, no parity, hardware (RTS/CTS) flow control
- **Telnet** - telnet server is running on 23 TCP port by default
- **SSH** - SSH (secure shell) server is running on 22 TCP port by default (available only if security package is installed)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **MAC Telnet** - MikroTik MAC Telnet potocol server is by default enabled on all Ethernet-like interfaces
- **Winbox** - Winbox is a RouterOS remote administration GUI for Windows, that use 8290 TCP port (or 8291 if security package is installed). It may also connect routers by their MAC addresses

Router may be managed through the following interfaces (note that until a valid IP configuration is enteres, telnet and SSH connections are not possible):

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Basic Setup Guide

Document revision 1.1 (Wed Sep 14 18:08:33 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Related Documents
  Description
Setting up MikroTik RouterOS™
  Description
  Notes
Logging into the MikroTik Router
  Description
Adding Software Packages
  Description
Navigating The Terminal Console
  Description
  Notes
Basic Configuration Tasks
  Description
  Notes
Setup Command
  Description
  Configure IP address on router, using the Setup command
Basic Examples
  Example
  Viewing Routes
  Adding Default Routes
  Testing the Network Connectivity
Advanced Configuration Tasks
  Description
  Application Example with Masquerading
  Example with Bandwidth Management
  Example with NAT

General Information

Summary

MikroTik RouterOS™ is independent Linux-based Operating System for IA-32 routers and thinrouters. It does not require any additional components and has no software prerequirements. It is designed with easy-to-use yet powerful interface allowing network administrators to deploy network structures and functions, that would require long education elsewhere simply by following the Reference Manual (and even without it).

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Related Documents

- ***Software Package Management***
- ***Device Driver List***
- ***License Management***
- ***Ping***
- ***Bandwidth Control***
-
- ***WinBox***
- ***Installing RouterOS with NetInstall***
- ***Installing RouterOS with CD-Install***
- ***Installing RouterOS with Floppies***

Description

MikroTik RouterOS™ turns a standard PC computer into a powerful network router. Just add standard network PC interfaces to expand the router capabilities. Remote control with easy real-time Windows application (WinBox)

- Advanced Quality of Service control with burst support
- Stateful firewall with P2P protocol filtering, tunnels and IPsec
- STP bridging with filtering capabilities
- WDS and Virtual AP features
- HotSpot for Plug-and-Play access
- RIP, OSPF, BGP routing protocols
- Gigabit Ethernet ready
- V.35, X.21, T1/E1 synchronous support
- async PPP with RADUIS AAA
- IP Telephony
- remote winbox GUI admin
- telnet/ssh/serial console admin
- real-time configuration and monitoring
- and much more (please see the Specifications Sheet)

The Guide describes the basic steps of installing and configuring a dedicated PC router running MikroTik RouterOS™.

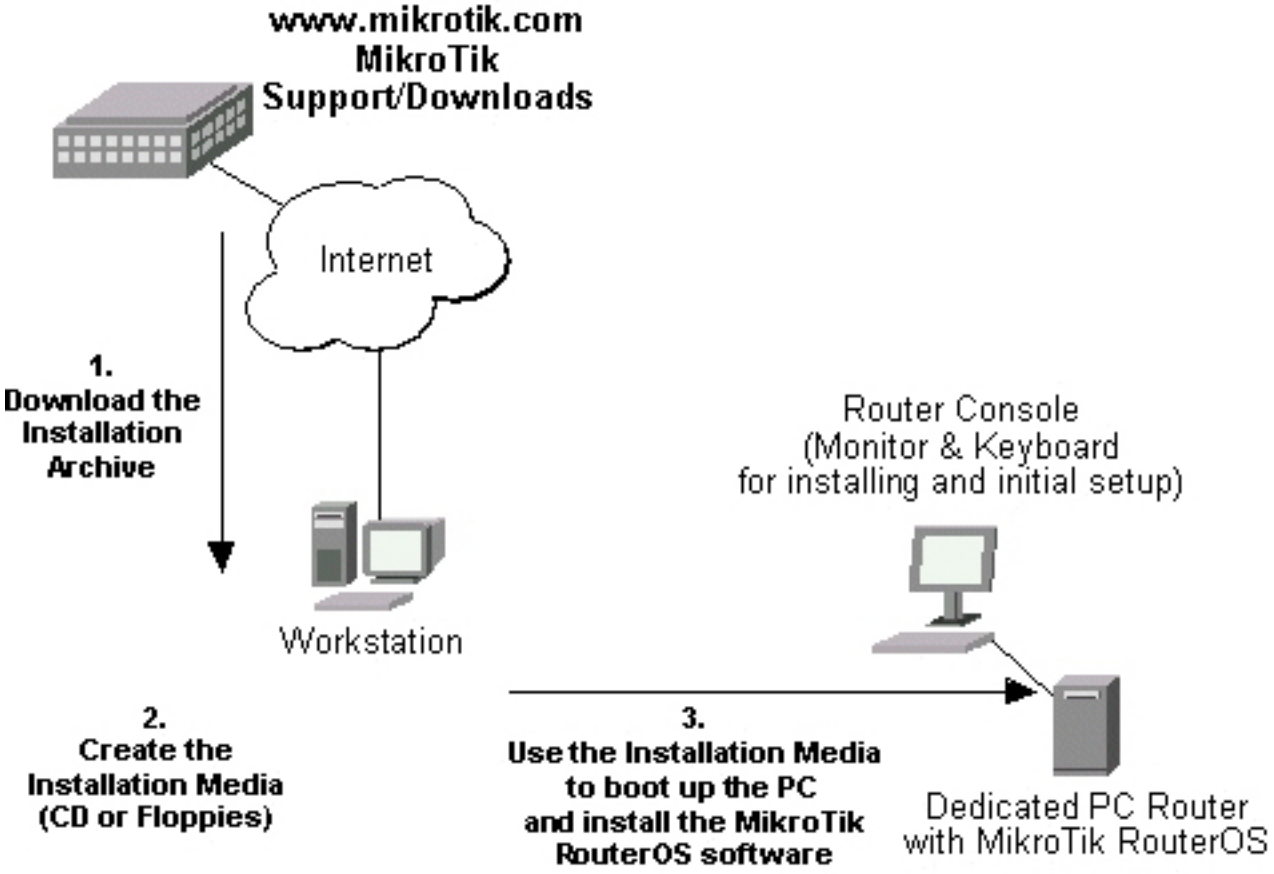
Setting up MikroTik RouterOS™

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

Downloading and Installing the MikroTik RouterOS™

The download and installation process of the MikroTik RouterOS™ is described in the following diagram:



1. Download the basic installation archieve file.
   Depending on the desired media to be used for installing the MikroTik RouterOS™ please chose one of the following archive types for downloading:
   - **ISO image** - of the installation CD, if you have a CD writer for creating CDs. The ISO image is in the MTcdimage_v2-9-x_dd-mmm-yyyy_(build_z).zip archive file containing a bootable CD image. The CD will be used for booting up the dedicated PC and installing the MikroTik RouterOS™ on its hard-drive or flash-drive.
   - **Netinstall** - if you want to install RouterOS over a LAN with one floppy boot disk, or alternatively using PXE or EtherBoot option supported by some network interface cards, that allows truly networked installation. Netinstall program works on Windows 95/98/NT4/2K/XP.
   - **MikroTik Disk Maker** - if you want to create 3.5" installation floppies. The Disk Maker is a self-extracting archive DiskMaker_v2-9-x_dd-mmm-yyyy_(build_z).exe file, which should be run on your Windows 95/98/NT4/2K/XP workstation to create the installation floppies. The installation floppies will be used for booting up the dedicated PC and installing the MikroTik RouterOS™ on its hard-drive or flash-drive.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

2. Create the installation media.
   Use the appropriate installation archive to create the Installation CD or floppies.
   - For the CD, write the ISO image onto a blank CD.
   - For the floppies, run the Disk Maker on your Windows workstation to create the installation floppies. Follow the instructions and insert the floppies in your FDD as requested, label them as Disk 1,2,3, etc.

3. Install the MikroTik RouterOS™ software.
   Your dedicated PC router hardware should have:

- **CPU and motherboard** - advanced 4th generation (core frequency 100MHz or more), 5th generation (Intel Pentium, Cyrix 6X86, AMD K5 or comparable) or newer uniprocessor Intel IA-32 (i386) compatible (multiple processors are not supported)
- **RAM** - minimum 64 MiB, maximum 1 GiB; 64 MiB or more recommended
- **Hard Drive/Flash** - standard ATA interface controller and drive (SCSI and USB controllers and drives are not supported; RAID controllers that require additional drivers are not supported) with minimum of 64 Mb space

Hardware needed for installation time only
Depending on installation method chosen the router must have the following hardware:

- **Floppy-based installation** - standard AT floppy controller and 3.5" disk drive connected as the first floppy disk drive (A); AT, PS/2 or USB keyboard; VGA-compatible video controller card and monitor
- **CD-based installation** - standard ATA/ATAPI interface controller and CD drive supporting "El Torito" bootable CDs (you might need also to check if the router's BIOS supports booting from this type of media; if El Torito is not supported by the BIOS, you can still boot up from the CD using Smart Boot Manager Floppy); AT, PS/2 or USB keyboard; VGA-compatible video controller card and monitor
- **Floppy-based network installation** - standard AT floppy controller and 3.5" disk drive connected as the first floppy disk drive (A); PCI Ethernet network interface card supported by MikroTik RouterOS (see the Device Driver List for the list)
- **Full network-based installation** - PCI Ethernet network interface card supported by MikroTik RouterOS (see the Device Driver List for the list) with PXE or EtherBoot extension booting ROM (you might need also to check if the router's BIOS supports booting from network)

Note that if you use Netinstall, you can license the software during the installation procedure (the next point of this section describes how to do it).
Boot up your dedicated PC router from the Installation Media you created and follow the instructions on the console screen while the HDD is reformatted and MikroTik RouterOS installed on it. After successful installation please remove the installation media from your CD or floppy disk drive and hit 'Enter' to reboot the router.

4. License the software.
   When booted, the software allows you to use all its features for 24 hours (note that you can pause the countdown by shutting down the router). If the license key will not be entered during this period of time, the router will become unusable, and will need a complete reinstallation.
   RouterOS licensing scheme is based on software IDs. To license the software, you must know the software ID. It is shown during installation procedures, and also you can get it from system console or Winbox. To get the software ID from system console, type: **/system license print** (note that you must first log in the router; by default there is user **admin** with no password

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

(just press [Enter] key when prompted for password)). See sections below on basic configuration of your router

Once you have the ID, you can obtain a license:

- You should have an account on our account server. If you do not have an account at www.mikrotik.com, just press the 'New' button on the upper right-hand corner of the MikroTik's web page to create your account
- Choose the appropriate licence level that meets your needs. Please see the ***License Manual*** or the ***Software price list***. Note that there is a free license with restricted features (no time limitation)
- There are different methods how to get a license from the account server:
  1. Enter the software ID in the account server, and get the license key by e-mail. You can upload the file received on the router's FTP server, or drag-and-drop it into opened Winbox window
  2. You can open the file with a text editor, and copy the contents. Then paste the text into system console (in any menu - you just should be logged in), or into System->License window of Winbox
  3. If the router has Internet connection, you can obtain the license directly from within it. The commands are described in the ***License Manual***. Note that you must have **Allow to use my account in netinstall** option enabled for your account. You can set it by following **change user information** link on the main screen of the account server.

Notes

The hard disk will be entirely reformatted during the installation and all data on it will be lost!

You can move the hard drive with MikroTik RouterOS installed to a new hardware without loosing a license, but you cannot move the RouterOS to a different hard drive without purchasing an another license (except hardware failure situations). For additional information write to ***key-support@mikrotik.com***.

Note! Do not use MS-DOS format command or other disk format utilities to reinstall your MikroTik router! This will cause the Software-ID to change, so you will need to buy another license in order to get MikroTik RouterOS running.

Logging into the MikroTik Router

Description

Normally you connect to the router by IP addresses with any telnet or SSH client software (a simple text-mode telnet client is usually called telnet and is distributed together with almost any OS). You can also use graphical configuration tool for Windows (also can be run in Linux using Wine) called Winbox. To get Winbox, connect to the router's IP address with a web browser, and follow the link to download winbox.exe from the router.

MAC-telnet is used to connect to a router when there is no other way to connect to it remotely if the

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

router has no IP address or in case of misconfigured firewall. MAC-telnet can only be used from the same broadcast domain (so there should be no routers in between) as any of the router's enabled interfaces (you can not connect to a disabled interface). MAC-telnet program is a part of the Neighbor Viewer. Download it from www.mikrotik.com, unpack both files contained in the archive to the same directory, and run NeighborViewer.exe. A list of MikroTik routers working in the same broadcast domain will be showed double-click the one you need to connect to. Note that Winbox is also able to connect to routers by their MAC addresses, and has the discovery tool built-in.

You can also connect to the router using a standard DB9 serial null-modem cable from any PC. Default settings of the router's serial port are 9600 bits/s (for RouterBOARD 500 series - 115200 bits/s), 8 data bits, 1 stop bit, no parity, hardware (RTS/CTS) flow control. Use terminal emulation program (like HyperTerminal or SecureCRT in Windows, or minicom in UNIX/Linux) to connect to the router. The router will beep twice when booted up, and you should see the login prompt shortly before that (check cabling and serial port settings if you do not see anything in the terminal window).

When logging into the router via terminal console, you will be presented with the MikroTik RouterOS™ login prompt. Use 'admin' and no password (hit [Enter]) for logging in the router for the first time, for example:

```
MikroTik v2.9
Login: admin
Password:
```

The password can be changed with the **/password** command.

```
[admin@MikroTik] > password
old password:
new password: ************
retype new password: ************
[admin@MikroTik] >
```

Adding Software Packages

Description

The basic installation comes only with the **system** package. This includes basic IP routing and router administration. To have additional features such as IP Telephony, OSPF, wireless and so on, you will need to ***download*** additional software packages.

The additional software packages should have **the same version** as the system package. If not, the package won't be installed. Please consult the MikroTik RouterOS™ Software Package Installation and Upgrading Manual for more detailed information about installing additional software packages.

To upgrade the router packages, simply upload the packages to the router via ftp, using the binary transfer mode. After you have uploaded the packages, reboot the router, and the features that are provided by those packages will be available (regarding your license type, of course).

Navigating The Terminal Console

Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Welcome Screen and Command Prompt

After logging into the router you will be presented with the MikroTik RouterOS™ Welcome Screen and command prompt, for example:

```
  MMM      MMM       KKK                          TTTTTTTTTTT      KKK
  MMMM    MMMM       KKK                          TTTTTTTTTTT      KKK
  MMM MMMM MMM  III  KKK  KKK  RRRRRR     OOOOOO      TTT     III  KKK  KKK
  MMM  MM  MMM  III  KKKKK     RRR  RRR  OOO  OOO     TTT     III  KKKKK
  MMM      MMM  III  KKK KKK   RRRRRR    OOO  OOO     TTT     III  KKK KKK
  MMM      MMM  III  KKK  KKK  RRR  RRR   OOOOOO      TTT     III  KKK  KKK

  MikroTik RouterOS 2.9 (c) 1999-2004       http://www.mikrotik.com/



Terminal xterm detected, using multiline input mode
[admin@MikroTik] >
```

The command prompt shows the identity name of the router and the current menu level, for example:

```
[admin@MikroTik] >
[admin@MikroTik] interface>
[admin@MikroTik] ip address>
```

Commands

The list of available commands at any menu level can be obtained by entering the question mark '?', for example:

```
[admin@MikroTik] >

log/ -- System logs
quit -- Quit console
radius/ -- Radius client settings
certificate/ -- Certificate management
special-login/ -- Special login users
redo -- Redo previously undone action
driver/ -- Driver management
ping -- Send ICMP Echo packets
setup -- Do basic setup of system
interface/ -- Interface configuration
password -- Change password
undo -- Undo previous action
port/ -- Serial ports
import -- Run exported configuration script
snmp/ -- SNMP settings
user/ -- User management
file/ -- Local router file storage.
system/ -- System information and utilities
queue/ -- Bandwidth management
ip/ -- IP options
tool/ -- Diagnostics tools
ppp/ -- Point to Point Protocol
routing/ -- Various routing protocol settings
export  --

[admin@MikroTik] >

[admin@MikroTik] ip>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
.. -- go up to root
service/ -- IP services
socks/ -- SOCKS version 4 proxy
arp/ -- ARP entries management
upnp/ -- Universal Plug and Play
dns/ -- DNS settings
address/ -- Address management
accounting/ -- Traffic accounting
the-proxy/ --
vrrp/ -- Virtual Router Redundancy Protocol
pool/ -- IP address pools
packing/ -- Packet packing settings
neighbor/ -- Neighbors
route/ -- Route management
firewall/ -- Firewall management
dhcp-client/ -- DHCP client settings
dhcp-relay/ -- DHCP relay settings
dhcp-server/ -- DHCP server settings
hotspot/ -- HotSpot management
ipsec/ -- IP security
web-proxy/ -- HTTP proxy
export --

[admin@MikroTik] ip>
```

The list of available commands and menus has short descriptions next to the items. You can move to the desired menu level by typing its name and hitting the [Enter] key, for example:

```
[admin@MikroTik] >                  | Base level menu
[admin@MikroTik] > driver           | Enter 'driver' to move to the driver
                                    | level menu
[admin@MikroTik] driver> /          | Enter '/' to move to the base level menu
                                    | from any level
[admin@MikroTik] > interface        | Enter 'interface' to move to the
                                    | interface level menu
[admin@MikroTik] interface> /ip     | Enter '/ip' to move to the IP level menu
                                    | from any level
[admin@MikroTik] ip>                |
```

A command or an argument does not need to be completed, if it is not ambiguous. For example, instead of typing **interface** you can type just **in** or **int**. To complete a command use the [Tab] key. Note that the completion is optional, and you can just use short command and parameter names

The commands may be invoked from the menu level, where they are located, by typing its name. If the command is in a different menu level than the current one, then the command should be invoked using its full (absolute) or relative path, for example:

```
[admin@MikroTik] ip route> print                | Prints the routing table
[admin@MikroTik] ip route> .. address print     | Prints the IP address table
[admin@MikroTik] ip route> /ip address print    | Prints the IP address table
```

The commands may have arguments. The arguments have their names and values. Some commands, may have a required argument that has no name.

Summary on executing the commands and navigating the menus

Command	Action
command [Enter]	**Executes the command**
[?]	**Shows the list of all available commands**

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

command [?]	**Displays help on the command and the list of arguments**
command argument [?]	**Displays help on the command's argument**
[Tab]	**Completes the command/word. If the input is ambiguous, a second [Tab] gives possible options**
/	**Moves up to the base level**
/command	**Executes the base level command**
..	**Moves up one level**
""	**Specifies an empty string**
"word1 word2"	**Specifies a string of 2 words that contain a space**

You can abbreviate names of levels, commands and arguments.

For the IP address configuration, instead of using the **address** and **netmask** arguments, in most cases you can specify the address together with the number of true bits in the network mask, i.e., there is no need to specify the **netmask** separately. Thus, the following two entries would be equivalent:

```
/ip address add address 10.0.0.1/24 interface ether1
/ip address add address 10.0.0.1 netmask 255.255.255.0 interface ether1
```

Notes

You must specify the size of the network mask in the address argument, even if it is the 32-bit subnet, i.e., use **10.0.0.1/32** for `address=10.0.0.1 netmask=255.255.255.255`

Basic Configuration Tasks

Description

Interface Management

Before configuring the IP addresses and routes please check the **/interface** menu to see the list of available interfaces. If you have Plug-and-Play cards installed in the router, it is most likely that the device drivers have been loaded for them automatically, and the relevant interfaces appear on the **/interface print** list, for example:

```
[admin@MikroTik] interface> print
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE    MTU
 0  R ether1                      ether           0          0          1500
 1  R ether2                      ether           0          0          1500
 2 X  wavelan1                    wavelan         0          0          1500
 3 X  prism1                      wlan            0          0          1500
[admin@MikroTik] interface>
```

The interfaces need to be enabled, if you want to use them for communications. Use the **/interface**

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

enable name command to enable the interface with a given name or number, for example:

```
[admin@MikroTik] interface> print
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE    MTU
 0 X  ether1                      ether           0          0          1500
 1 X  ether2                      ether           0          0          1500
[admin@MikroTik] interface> enable 0
[admin@MikroTik] interface> enable ether2
[admin@MikroTik] interface> print
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE    MTU
 0  R ether1                      ether           0          0          1500
 1  R ether2                      ether           0          0          1500
[admin@MikroTik] interface>
```

The interface name can be changed to a more descriptive one by using **/interface set** command:

```
[admin@MikroTik] interface> set 0 name=Local; set 1 name=Public
[admin@MikroTik] interface> print
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE    MTU
 0  R Local                       ether           0          0          1500
 1  R Public                      ether           0          0          1500
[admin@MikroTik] interface>
```

Notes

The device drivers for NE2000 compatible ISA cards need to be loaded using the **add** command under the /drivers menu. For example, to load the driver for a card with IO address 0x280 and IRQ 5, it is enough to issue the command:

```
[admin@MikroTik] driver> add name=ne2k-isa io=0x280
[admin@MikroTik] driver> print
Flags: I - invalid, D - dynamic
 #   DRIVER                              IRQ IO       MEMORY   ISDN-PROTOCOL
 0 D RealTek 8139
 1 D Intel EtherExpressPro
 2 D PCI NE2000
 3   ISA NE2000                          280
 4   Moxa C101 Synchronous                            C8000
[admin@MikroTik] driver>
```

There are some other drivers that should be added manually. Please refer to the respective manual sections for the detailed information on how drivers are to be loaded.

Setup Command

Command name: ***/setup***

Description

The initial setup of the router can be done by using the **/setup** command which offers the following configuration:

- reset all router configuration
- load interface driver
- configure ip address and gateway
- setup dhcp client

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- setup dhcp server
- setup pppoe client
- setup pptp client

Configure IP address on router, using the Setup command

Execute the **/setup** command from command line:

```
[admin@MikroTik] > setup
  Setup uses Safe Mode. It means that all changes that are made during setup
are reverted in case of error, or if [Ctrl]+[C] is used to abort setup. To keep
changes exit setup using the [X] key.

[Safe Mode taken]
  Choose options by pressing one of the letters in the left column, before
dash. Pressing [X] will exit current menu, pressing Enter key will select the
entry that is marked by an '*'. You can abort setup at any time by pressing
[Ctrl]+[C].
Entries marked by '+' are already configured.
Entries marked by '-' cannot be used yet.
Entries marked by 'X' cannot be used without installing additional packages.
   r - reset all router configuration
 + l - load interface driver
 * a - configure ip address and gateway
   d - setup dhcp client
   s - setup dhcp server
   p - setup pppoe client
   t - setup pptp client
   x - exit menu
your choice [press Enter to configure ip address and gateway]: a
```

To configure IP address and gateway, press **a** or [Enter], if the **a** choice is marked with an asterisk symbol ('*').

```
  * a - add ip address
  - g - setup default gateway
    x - exit menu
your choice [press Enter to add ip address]: a
```

Choose **a** to add an IP address. At first, setup will ask you for an interface to which the address will be assigned. If the setup offers you an undesirable interface, erase this choice, and press the [Tab] key twice to see all available interfaces. After the interface is chosen, assign IP address and network mask on it:

```
your choice: a
enable interface:
ether1  ether2  wlan1
enable interface: ether1
ip address/netmask: 10.1.0.66/24
#Enabling interface
/interface enable ether1
#Adding IP address
/ip address add address=10.1.0.66/24 interface=ether1 comment="added by setup"
 + a - add ip address
 * g - setup default gateway
   x - exit menu
your choice: x
```

Basic Examples

Example

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Assume you need to configure the MikroTik router for the following network setup:

In the current example we use two networks:

- The local LAN with network address 192.168.0.0 and 24-bit netmask: 255.255.255.0. The router's address is 192.168.0.254 in this network
- The ISP's network with address 10.0.0.0 and 24-bit netmask 255.255.255.0. The router's address is 10.0.0.217 in this network

The addresses can be added and viewed using the following commands:

```
[admin@MikroTik] ip address> add address 10.0.0.217/24 interface Public
[admin@MikroTik] ip address> add address 192.168.0.254/24 interface Local
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.217/24      10.0.0.217      10.0.0.255      Public
  1   192.168.0.254/24   192.168.0.0     192.168.0.255   Local
[admin@MikroTik] ip address>
```

Here, the network mask has been specified in the value of the address argument. Alternatively, the argument 'netmask' could have been used with the value '255.255.255.0'. The network and broadcast addresses were not specified in the input since they could be calculated automatically.

Please **note** that the addresses assigned to different interfaces of the router should belong to different networks.

Viewing Routes

You can see two dynamic (D) and connected (C) routes, which have been added automatically when the addresses were added in the example above:

```
[admin@MikroTik] ip route> print
Flags:  A - active, X - disabled, I - invalid, D - dynamic, C - connect,
        S - static, r - rip, b - bgp, o - ospf, d - dynamic
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0 ADC 192.168.0.0/24     r 0.0.0.0          0        Local
    1 ADC 10.0.0.0/24        r 0.0.0.0          0        Public
[admin@MikroTik] ip route> print detail
Flags: A - active, X - disabled, I - invalid, D - dynamic, C - connect,
       S - static, r - rip, b - bgp, o - ospf, d - dynamic
 0 ADC dst-address=192.168.0.0/24 prefsrc=192.168.0.254 interface=Local scope=10

 1 ADC dst-address=10.0.0.0/24 prefsrc=10.0.0.217 interface=Public scope=10

[admin@MikroTik] ip route>
```

These routes show, that IP packets with destination to 10.0.0.0/24 would be sent through the interface Public, whereas IP packets with destination to 192.168.0.0/24 would be sent through the interface Local. However, you need to specify where the router should forward packets, which have destination other than networks connected directly to the router.

Adding Default Routes

In the following example the `default route` (destination 0.0.0.0 (any), netmask 0.0.0.0 (any)) will be added. In this case it is the ISP's gateway 10.0.0.1, which can be reached through the interface **Public**

```
[admin@MikroTik] ip route> add gateway=10.0.0.1
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #     DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0 ADC 192.168.0.0/24                                  Local
    1 ADC 10.0.0.0/24                                    Public
    2 A S 0.0.0.0/0          r 10.0.0.1         0      Public
[admin@MikroTik] ip route>
```

Here, the default route is listed under #2. As we see, the gateway 10.0.0.1 can be reached through the interface 'Public'. If the gateway was specified incorrectly, the value for the argument 'interface' would be unknown.

Notes

You cannot add two routes to the same destination, i.e., destination-address/netmask! It applies to the default routes as well. Instead, you can enter multiple gateways for one destination. For more information on IP routes, please read the ***Routes, Equal Cost Multipath Routing, Policy Routing*** manual.

If you have added an unwanted static route accidentally, use the **remove** command to delete the unneeded one. You will not be able to delete dynamic (DC) routes. They are added automatically and represent routes to the networks the router connected directly.

Testing the Network Connectivity

From now on, the **/ping** command can be used to test the network connectivity on both interfaces. You can reach any host on both connected networks from the router.

How the **/ping** command works:

```
[admin@MikroTik] ip route> /ping 10.0.0.4
10.0.0.4 64 byte ping: ttl=255 time=7 ms
10.0.0.4 64 byte ping: ttl=255 time=5 ms
10.0.0.4 64 byte ping: ttl=255 time=5 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 5/5.6/7 ms
[admin@MikroTik] ip route>
[admin@MikroTik] ip route> /ping 192.168.0.1
192.168.0.1 64 byte ping: ttl=255 time=1 ms
192.168.0.1 64 byte ping: ttl=255 time=1 ms
192.168.0.1 64 byte ping: ttl=255 time=1 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 1/1.0/1 ms
[admin@MikroTik] ip route>
```

The workstation and the laptop can reach (ping) the router at its local address 192.168.0.254, If the router's address 192.168.0.254 is specified as the default gateway in the TCP/IP configuration of both the workstation and the laptop, then you should be able to ping the router:

```
C:\>ping 192.168.0.254
Reply from 192.168.0.254: bytes=32 time=10ms TTL=253
Reply from 192.168.0.254: bytes=32 time<10ms TTL=253
Reply from 192.168.0.254: bytes=32 time<10ms TTL=253

C:\>ping 10.0.0.217
Reply from 10.0.0.217: bytes=32 time=10ms TTL=253
Reply from 10.0.0.217: bytes=32 time<10ms TTL=253
Reply from 10.0.0.217: bytes=32 time<10ms TTL=253

C:\>ping 10.0.0.4
Request timed out.
Request timed out.
Request timed out.
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Notes

You cannot access anything beyond the router (network 10.0.0.0/24 and the Internet), unless you do the one of the following:

- Use source network address translation (masquerading) on the MikroTik router to 'hide' your private LAN 192.168.0.0/24 (see the information below), or
- Add a static route on the ISP's gateway 10.0.0.1, which specifies the host 10.0.0.217 as the gateway to network 192.168.0.0/24. Then all hosts on the ISP's network, including the server, will be able to communicate with the hosts on the LAN

To set up routing, it is required that you have some knowledge of configuring TCP/IP networks. We strongly recommend that you obtain more knowledge, if you have difficulties configuring your network setups.

Advanced Configuration Tasks

Description

Next will be discussed situation with 'hiding' the private LAN 192.168.0.0/24 'behind' one address 10.0.0.217 given to you by the ISP.

Application Example with Masquerading

If you want to 'hide' the private LAN 192.168.0.0/24 'behind' one address 10.0.0.217 given to you by the ISP, you should use the source network address translation (masquerading) feature of the MikroTik router. Masquerading is useful, if you want to access the ISP's network and the Internet appearing as all requests coming from the host 10.0.0.217 of the ISP's network. The masquerading will change the source IP address and port of the packets originated from the network 192.168.0.0/24 to the address 10.0.0.217 of the router when the packet is routed through it.

Masquerading conserves the number of global IP addresses required and it lets the whole network use a single IP address in its communication with the world.

To use masquerading, a source NAT rule with action 'masquerade' should be added to the firewall configuration:

```
[admin@MikroTik] ip firewall nat> add chain=srcnat action=masquerade
out-interface=Public
[admin@MikroTik] ip firewall nat> print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=srcnat out-interface=Public action=masquerade
```

Notes

Please consult ***Network Address Translation*** for more information on masquerading.

Example with Bandwidth Management

Assume you want to limit the bandwidth to 128kbps on downloads and 64kbps on uploads for all hosts on the LAN. Bandwidth limitation is done by applying queues for outgoing interfaces

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

regarding the traffic flow. It is enough to add a single queue at the MikroTik router:

```
[admin@MikroTik] queue simple> add max-limit=64000/128000 interface=Local
[admin@MikroTik] queue simple> print
Flags: X - disabled, I - invalid, D - dynamic
 0    name="queue1" target-address=0.0.0.0/0 dst-address=0.0.0.0/0
      interface=Local queue=default/default priority=8 limit-at=0/0
      max-limit=64000/128000 total-queue=default
[admin@MikroTik] queue simple>
```

Leave all other parameters as set by default. The limit is approximately 128kbps going to the LAN (download) and 64kbps leaving the client's LAN (upload).

Example with NAT

Assume we have moved the server in our previous examples from the public network to our local one:

The server's address is now 192.168.0.4, and we are running web server on it that listens to the TCP port 80. We want to make it accessible from the Internet at address:port 10.0.0.217:80. This can be done by means of Static Network Address translation (NAT) at the MikroTik Router. The Public address:port 10.0.0.217:80 will be translated to the Local address:port 192.168.0.4:80. One destination NAT rule is required for translating the destination address and port:

```
[admin@MikroTik] ip firewall nat> add chain=dstnat action=dst-nat protocol=tcp
dst-address=10.0.0.217/32
                                  dst-port=80 to-addresses=192.168.0.4
[admin@MikroTik] ip firewall nat> pr
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=dstnat dst-address=10.0.0.217/32 protocol=tcp dst-port=80
     action=dst-nat to-addresses=192.168.0.4 to-ports=0-65535
```

Notes

Please consult ***Network Address Translation*** for more information on Network Address Translation.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Installing RouterOS with CD-Install

Document revision 1.2 (Tue Jul 13 13:06:16 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
CD-Install
  Description

CD-Install

Description

To install the RouterOS using a CD you will need a CD-writer and a blank CD. Burn the CD-image (an **.iso** file) to a CD. The archive with image can be downloaded ***here***.

Follow the instructions to install RouterOS using CD-Install:

1. After downloading the CD image from ***www.mikrotik.com*** you will have an ISO file on your computer:
2. Open a CD Writing software, like ***Ahead NERO*** as in this example:
3. In the program, choose **Burn Image** entry from the **Recorder** menu (there should be similary named option in all major CD burning programs):
4. Select the recently extracted ISO file and click **Open:**
5. Finally, click **Burn** button:
6. Set the first boot device to CDROM in router's BIOS.
7. After booting from CD you will see a menu where to choose packages to install:

```
                Welcome to MikroTik Router Software installation

Move around menu using 'p' and 'n' or arrow keys, select with 'spacebar'.
Select all with 'a', minimum with 'm'. Press 'i' to install locally or 'r' to
install remote router or 'q' to cancel and reboot.

  [X] system              [ ] isdn                [ ] synchronous
  [X] ppp                 [ ] lcd                 [ ] telephony
  [X] dhcp                [ ] ntp                 [ ] ups
  [X] advanced-tools      [ ] radiolan            [ ] web-proxy
  [ ] arlan               [ ] routerboard         [ ] wireless
  [ ] gps                 [X] routing
  [ ] hotspot             [X] security
```

   Follow the instructions, select needed packages, and press 'i' to install the software.

8. You will be asked for 2 questions:

```
Warning: all data on the disk will be erased!

Continue? [y/n]
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Press [Y] to continue or [N] to abort the installation.

```
Do you want to keep old configuration? [y/n]:
```

You should choose whether you want to keep old configuration (press [Y]) or to erase the configuration permanently (press [N]) and continue without saving it. For a fresh installation, press [N].

```
Creating partition...
Formatting disk...
```

The system will install selected packages. After that you will be prompted to press 'Enter'. Before doing that, remove the CD from your CD-Drive:

```
Software installed.
Press ENTER to reboot
```

Note: after the installation you will have to enter the Software key. See ***this*** manual how to do it.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Installing RouterOS with Floppies

Document revision 1.2 (Tue Jul 13 13:06:16 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
Floppy Install
  Description
  para

Floppy Install

Description

Another way to install the RouterOS is using floppies. You will need 9 floppies to install the software (this includes only the system package).

1. Download the archive ***here***. Extract it and run FloppyMaker.exe.
   Read the licence agreement and press 'Yes' to continue.
2. After pressing 'Yes', you are introduced to useful information about RouterOS:
   Press 'Continue' button to continue or 'Exit' to leave the installation.
3. You are prompted to insert disk #1 into the floppy drive:
   Insert a blank floppy into the drive and start the copying process. Pressing 'Skip Floppy' will skip the process to next floppy (useful in case you already have some floppies copied). Proceed with next floppies until the following dialog occurs:
4. Set the dedicated computer to boot from floppy device, insert the disk #1 and boot the computer. When it will process the first floppy, it will ask for the second, until all floppies are processed.

Note: after the installation you will have to enter the Software key. See ***this*** manual how to do it.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Installing RouterOS with NetInstall

Document revision 1.3 (Mon Jul 19 12:58:25 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
NetInstall
  Description

NetInstall

Description

NetInstall is a program that allows you to install MikroTIK RouterOS on a dedicated PC or RouterBoard via Ethernet network. All you need is a blank floppy or an Ethernet device that supports PXE, an Ethernet network between workstation and dedicated computer, and a serial null-modem console cable (for RouterBoard 200).

NetInstall Program Parameters

The program runs on Windows 95/98/ME/NT/2000/XP platforms.

Netinstall parameters:

- **Routers/Drives** - in this list you can see all the devices waiting for installation.
- **Software ID** - a unique ID that is generated for licensing purposes.
- **Key** - a key that is generated for the Software ID. When you purchase a license, you get a **key** file. Click the **Browse...** button next to the **key** field to select your **key** file.
- **Get Key...** - obtain software key from MikroTIK server:
  - **Software ID** - ID for which the key will be generated (depending on the license level).
  - **Username** - client's username in the Account data base.
  - **Password** - client's password.
  - **Level** - license level of RouterOS.
  - **Debit key** - a key that you have paid for, but haven't generated yet.
  - **Debit money** - money that you have on your account. To add money to your account, use the 'add debit' link in the account server.
  - **Credit key** - a key that you can take now, but pay later.
  - **Credit money** - paying with credit money allows you to get your keys now and pay for them later.
- **Keep old configuration** - used for reinstalling the software. If checked, the old configuration on the router will not be overwritten, otherwise it will be lost.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **IP address/mask** - address with subnet mask that will be assigned to **ether1** interface after the packages are installed.
- **Gateway** - specifies the default gateway (static route).
- **Baud rate** - this baud rate will be set for serial console (bps).
- **Configure script** - a RouterOS script to execute after the package installation.
- **Make floppy** - make a bootable NetInstall floppy.
- **Net booting** - opens the Network Booting Settings window. Enter an IP address from your local network. This address will be temporarily assigned to the computer where RouterOS will be installed on.
- **Install** - installs the RouterOS on a computer.
- **Cancel** - cancel the installation.
- **Sets** - an entry in this list represents the choice of packages selected to install from a directory. If you want to make your own set, browse for a folder that contains packages (*.npk files), select needed packages in the list, and press the **Save set** button.
- **From** - type the directory where your packages are stored or press the **Browse...** button to select the directory.
- **Select all** - selects all packages in the list
- **Select none** - unselects all packages in the list

Note: some of the **Get key...** parameters could not be available for all account types.

NetInstall Example

This example shows step-by-step instructions how to install the software on a RouterBoard 200.

1. Connect the routerboard to a switch (or a hub) as it is shown in the diagram using ether1 interface (on RouterBoard 230 it is next to the RS-232 interface):
2. Run NetInstall program on your workstation (you can download it ***here***. It is necessary to extract the packages (*.npk files) on your hard drive.
   NetInstall v1.10
3. Enter the Boot Server Client's IP address. Use an address from a network to which belongs your NIC (in this case **172.16.0.0/24**). This IP address will be temporarily assigned to the routerboard.
4. Set the RouterBoard to boot from Ethernet interface. To do this, enter the RouterBoard BIOS (press any key when prompted):

```
RouterBIOS v1.3.0 MikroTik (tm) 2003-2004

RouterBOARD 230 (CPU revision B1)
CPU frequency: 266 MHz
  Memory size:  64 MB

Press any key within 1 second to enter setup.
```

You will see a list of available commands. To set up the boot device, press the 'o' key:

```
 RouterBIOS v1.3.0
What do you want to configure?
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
   d - boot delay
   k - boot key
   s - serial console
   l - debug level
   o - boot device
   b - beep on boot
   v - vga to serial
   t - ata translation
   p - memory settings
   m - memory test
   u - cpu mode
   f - pci back-off
   r - reset configuration
   g - bios upgrade through serial port
   c - bios license information
   x - exit setup
your choice: o - boot device
```

Press the 'e' key to make the RouterBoard to boot from Ethernet interface:

```
Select boot device:
 * i - IDE
   e - Etherboot
   1 - Etherboot (timeout 15s), IDE
   2 - Etherboot (timeout  1m), IDE
   3 - Etherboot (timeout  5m), IDE
   4 - Etherboot (timeout 30m), IDE
   5 - IDE, try Etherboot first on next boot (15s)
   6 - IDE, try Etherboot first on next boot (1m)
   7 - IDE, try Etherboot first on next boot (5m)
   8 - IDE, try Etherboot first on next boot (30m)
your choice: e - Etherboot
```

When this is done, the RouterBoard BIOS will return to the first menu. Press the 'x' key to exit from BIOS. The router will reboot.

5. When booting up, the RouterBoard will try to boot from its Ethernet device. If successful, the Workstation will give to this RouterBoard an IP address, specified in Network Booting Settings. After this process, the RouterBoard will be waiting for installation.
   On the workstation, there will appear a new entry in Routers/Drives list:
   You can identify the router by MAC address in the list. Click on the desired entry and you will be able to configure ***installation parameters***.
   When done, press the Install button to install RouterOS.
6. When the installation process has finished, press 'Enter' on the console or 'Reboot' button in the NetInstall program. Remember to set the boot device back to IDE in the RouterBoard BIOS.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Configuration Management

Document revision 1.6 (Mon Sep 19 12:55:52 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Description
System Backup
  Description
  Command Description
  Example
  Example
The Export Command
  Description
  Command Description
  Example
The Import Command
  Description
  Command Description
  Example
Configuration Reset
  Description
  Command Description
  Notes
  Example

General Information

Summary

This manual introduces you with commands which are used to perform the following functions:

- system backup
- system restore from a backup
- configuration export
- configuration import
- system configuration reset

Description

The configuration backup can be used for backing up MikroTik RouterOS configuration to a binary file, which can be stored on the router or downloaded from it using FTP. The configuration restore can be used for restoring the router's configuration from a backup file.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

The configuration export can be used for dumping out MikroTik RouterOS configuration to the console screen or to a text (script) file, which can be downloaded from the router using FTP. The configuration import can be used to import the router configuration script from a text file.

System reset command is used to erase all configuration on the router. Before doing that, it might be useful to backup the router's configuration.

Note! In order to be sure that the backup will not fail, **system backup load** command must be used on the same computer with the same hardware where **system backup save** was done.

System Backup

Home menu level: ***/system backup***

Description

The **save** command is used to store the entire router configuration in a backup file. The file is shown in the **/file** submenu. It can be downloaded via ftp to keep it as a backup for your configuration.

To restore the system configuration, for example, after a **/system reset**, it is possible to upload that file via ftp and load that backup file using **load** command in **/system backup** submenu.

Command Description

load name=[filename] - Load configuration backup from a file

save name=[filename] - Save configuration backup to a file

Example

To save the router configuration to file **test**:

```
[admin@MikroTik] system backup> save name=test
Configuration backup saved
[admin@MikroTik] system backup>
```

To see the files stored on the router:

```
[admin@MikroTik] > file print
  # NAME                          TYPE         SIZE       CREATION-TIME
  0 test.backup                   backup       12567      sep/08/2004 21:07:50
[admin@MikroTik] >
```

Example

To load the saved backup file **test**:

```
[admin@MikroTik] system backup> load name=test
Restore and reboot? [y/N]: y
...
```

The Export Command

Command name: ***/export***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

The **export** command prints a script that can be used to restore configuration. The command can be invoked at any menu level, and it acts for that menu level and all menu levels below it. If the argument **from** is used, then it is possible to export only specified items. In this case **export** does not descend recursively through the command hierarchy. **export** also has the argument **file**, which allows you to save the script in a file on the router to retrieve it later via FTP.

Command Description

file=[filename] - saves the export to a file

from=[number] - specifies from which item to start to generate the export file

Example

```
[admin@MikroTik] > ip address print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   10.1.0.172/24      10.1.0.0        10.1.0.255      bridge1
 1   10.5.1.1/24        10.5.1.0        10.5.1.255      ether1
[admin@MikroTik] >
```

To make an export file:

```
[admin@MikroTik] ip address> export file=address
[admin@MikroTik] ip address>
```

To make an export file from only one item:

```
[admin@MikroTik] ip address> export file=address1 from=1
[admin@MikroTik] ip address>
```

To see the files stored on the router:

```
[admin@MikroTik] > file print
 # NAME                            TYPE         SIZE       CREATION-TIME
0  address.rsc                     script       315        dec/23/2003 13:21:48
1  address1.rsc                    script       201        dec/23/2003 13:22:57
[admin@MikroTik] >
```

To export the setting on the display use the same command without the **file** argument:

```
[admin@MikroTik] ip address> export from=0,1
# nov/13/2004 13:25:30 by RouterOS 2.9
# software id = MGJ4-MAN
#
/ ip address
add address=10.1.0.172/24 network=10.1.0.0 broadcast=10.1.0.255 \
    interface=bridge1 comment="" disabled=no
add address=10.5.1.1/24 network=10.5.1.0 broadcast=10.5.1.255 \
    interface=ether1 comment="" disabled=no
[admin@MikroTik] ip address>
```

The Import Command

Command name: ***/import***

Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

The root level command **/import [file_name]** restores the exported information from the specified file. This is used to restore configuration or part of it after a **/system reset** event or anything that causes configuration data loss.

Note that it is impossible to import the whole router configuration using this feature. It can only be used to import a part of configuration (for example, firewall rules) in order to spare you some typing.

Command Description

file=[filename] - loads the exported configuration from a file to router

Example

To load the saved export file use the following command:

```
[admin@MikroTik] > import address.rsc
Opening script file address.rsc
Script file loaded successfully
[admin@MikroTik] >
```

Configuration Reset

Command name: ***/system reset***

Description

The command clears all configuration of the router and sets it to the default including the login name and password ('admin' and no password), IP addresses and other configuration is erased, interfaces will become disabled. After the **reset** command router will reboot.

Command Description

reset - erases router's configuration

Notes

If the router has been installed using netinstall and had a script specified as the initial configuration, the reset command executes this script after purging the configuration. To stop it doing so, you will have to reinstall the router.

Example

```
[admin@MikroTik] > system reset
Dangerous! Reset anyway? [y/N]: n
action cancelled
[admin@MikroTik] >
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

FTP (File Transfer Protocol) Server

Document revision 2.3 (Fri Jul 08 15:52:48 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
File Transfer Protocol Server
  Description
  Property Description
  Command Description

General Information

Summary

MikroTik RouterOS implements File Transfer Protocol (FTP) server feature. It is intended to be used for software packages uploading, configuration script exporting and importing procedures, as well as for storing HotSpot servlet pages.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/file***
Standards and Technologies: ***FTP (RFC 959)***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***Configuration Management***

File Transfer Protocol Server

Home menu level: ***/file***

Description

MikroTik RouterOS has an industry standard FTP server feature. It uses ports 20 and 21 for communication with other hosts on the network.

Uploaded files as well as exported configuration or backup files can be accessed under /file menu. There you can delete unnecessary files from your router.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Authorization for FTP service uses router's system user account names and passwords.

Property Description

creation-time (*read-only: time*) - item creation date and time

name (*read-only: name*) - item name

size (*read-only: integer*) - package size in bytes

type (*read-only: file* | *directory* | *unknown* | *script* | *package* | *backup*) - item type

Command Description

print - shows a list of files stored - shows contents of files less that 4kb long - offers to edit file's contents with editor - sets the file's contents to 'content'

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

MAC Level Access (Telnet and Winbox)

Document revision 2.2 (Wed Oct 05 16:26:50 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
MAC Telnet Server
  Property Description
  Notes
  Example
MAC WinBox Server
  Property Description
  Notes
  Example
Monitoring Active Session List
  Property Description
  Example
MAC Telnet Client
  Example

General Information

Summary

MAC telnet is used to provide access to a router that has no IP address set. It works just like IP telnet. MAC telnet is possible between two MikroTik RouterOS routers only.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/tool, /tool mac-server***
Standards and Technologies: ***MAC Telnet***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***WinBox***
- ***Ping***
- ***MNDP***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

MAC Telnet Server

Home menu level: ***/tool mac-server***

Property Description

interface (*name* | *all*; default: **all**) - interface name to which the mac-server clients will connect

- **all** - all interfaces

Notes

There is an interface list in this submenu level. If you add some interfaces to this list, you allow MAC telnet to that interface. Disabled (**disabled=yes**) item means that interface is not allowed to accept MAC telnet sessions on that interface.

Example

To enable MAC telnet server on **ether1** interface only:

```
[admin@MikroTik] tool mac-server> print
Flags: X - disabled
 #   INTERFACE
 0   all
[admin@MikroTik] tool mac-server> remove 0
[admin@MikroTik] tool mac-server> add interface=ether1 disabled=no
[admin@MikroTik] tool mac-server> print
Flags: X - disabled
 #   INTERFACE
 0   ether1
[admin@MikroTik] tool mac-server>
```

MAC WinBox Server

Home menu level: ***/tool mac-server mac-winbox***

Property Description

interface (*name* | *all*; default: **all**) - interface name to which it is alowed to connect with Winbox using MAC-based protocol

- **all** - all interfaces

Notes

There is an interface list in this submenu level. If you add some interfaces to this list, you allow MAC Winbox to that interface. Disabled (**disabled=yes**) item means that interface is not allowed to accept MAC Winbox sessions on that interface.

Example

To enable MAC Winbox server on **ether1** interface only:

```
[admin@MikroTik] tool mac-server mac-winbox> print
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
Flags: X - disabled
 #   INTERFACE
 0   all
[admin@MikroTik] tool mac-server mac-winbox> remove 0
[admin@MikroTik] tool mac-server mac-winbox> add interface=ether1 disabled=no
[admin@MikroTik] tool mac-server mac-winbox> print
Flags: X - disabled
 #   INTERFACE
 0   ether1
[admin@MikroTik] tool mac-server mac-winbox>
```

Monitoring Active Session List

Home menu level: ***/tool mac-server sessions***

Property Description

interface (*read-only: name*) - interface to which the client is connected to

src-address (*read-only: MAC address*) - client's MAC address

uptime (*read-only: time*) - how long the client is connected to the server

Example

To see active MAC Telnet sessions:

```
[admin@MikroTik] tool mac-server sessions> print
 # INTERFACE SRC-ADDRESS       UPTIME
 0 wlan1     00:0B:6B:31:08:22 00:03:01
[admin@MikroTik] tool mac-server sessions>
```

MAC Telnet Client

Command name: ***/tool mac-telnet [MAC-address]***

Example

```
[admin@MikroTik] > /tool mac-telnet 00:02:6F:06:59:42
Login: admin
Password:
Trying 00:02:6F:06:59:42...
Connected to 00:02:6F:06:59:42

  MMM      MMM       KKK                          TTTTTTTTTTT      KKK
  MMMM    MMMM       KKK                          TTTTTTTTTTT      KKK
  MMM MMMM MMM  III  KKK  KKK  RRRRRR     OOOOOO      TTT     III  KKK  KKK
  MMM  MM  MMM  III  KKKKK     RRR  RRR  OOO  OOO     TTT     III  KKKKK
  MMM      MMM  III  KKK KKK   RRRRRR    OOO  OOO     TTT     III  KKK KKK
  MMM      MMM  III  KKK  KKK  RRR  RRR   OOOOOO      TTT     III  KKK  KKK

  MikroTik RouterOS 2.9 (c) 1999-2004             http://www.mikrotik.com/

Terminal linux detected, using multiline input mode
[admin@MikroTik] >
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Serial Console and Terminal

Document revision 2.1 (Wed Mar 03 16:12:49 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
Serial Console Configuration
  Description
Configuring Console
  Property Description
  Example
Using Serial Terminal
  Description
  Property Description
  Notes
  Example
Console Screen
  Description
  Property Description
  Notes
  Example

General Information

Summary

The Serial Console and Terminal are tools, used to communicate with devices and other systems that are interconnected via serial port. The serial terminal may be used to monitor and configure many devices - including modems, network devices (including MikroTik routers), and any device that can be connected to a serial (asynchronous) port.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/system, /system console, /system serial-terminal***
Standards and Technologies: ***RS-232***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

The Serial Console (managed side) feature allows configuring one serial port of the MikroTik router for access to the router's Terminal Console over the serial port. A special null-modem cable is required to connect the router's serial port with the workstation's or laptop's serial (COM) port. A terminal emulation program, e.g., HyperTerminal, should be run on the workstation. You can also use MikroTik RouterOS to connect to an another Serial Console (for example, on a Cisco router).

Several customers have described situations where the Serial Terminal (managing side) feature would be useful:

- in a mountaintop where a MikroTik wireless installation sits next to equipment (including switches and Cisco routers) that can not be managed in-band (by telnet through an IP network)
- monitoring weather-reporting equipment through a serial-console
- connection to a high-speed microwave modem that needed to be monitored and managed by a serial-console connection

With the serial-terminal feature of the MikroTik, up to 132 (and, maybe, even more) devices can be monitored and controlled

Serial Console Configuration

Description

A special null-modem cable should be used for connecting to the serial console. The Serial Console cabling diagram for DB9 connectors is as follows:

Router Side (DB9f)	Signal	Direction	Side (DB9f)
1, 6	**CD, DSR**	**IN**	**4**
2	**RxD**	**IN**	**3**
3	**TxD**	**OUT**	**2**
4	**DTR**	**OUT**	**1, 6**
5	**GND**	**-**	**5**
7	**RTS**	**OUT**	**8**
8	**CTS**	**IN**	**7**

Configuring Console

Home menu level: ***/system console***

Property Description

enabled (*yes* | *no*; default: **no**) - whether serial console is enabled or not

free (*read-only: text*) - console is ready for use

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

port (*name*; default: **serial0**) - which port should the serial terminal listen to

term (*text*) - name for the terminal

used (*read-only: text*) - console is in use

vcno (*read-only: integer*) - number of virtual console - [Alt]+[F1] represents '1', [Alt]+[F2] - '2', etc.

wedged (*read-only: text*) - console is currently not available

Example

To enable Serial Console with terminal name **MyConsole**:

```
[admin@MikroTik] system console> set 0 disabled=no term=MyConsole
[admin@MikroTik] system console> print
Flags: X - disabled, W - wedged, U - used, F - free
 #   PORT    VCNO       TERM
 0 F serial0            MyConsole
 1 W          1          linux
 2 W          2          linux
 3 W          3          linux
 4 W          4          linux
 5 W          5          linux
 6 W          6          linux
 7 W          7          linux
 8 W          8          linux
[admin@MikroTik] system console>
```

To check if the port is available or used (parameter **used-by**):

```
[admin@MikroTik] system serial-console> /port print detail
  0 name=serial0 used-by=Serial Console baud-rate=9600 data-bits=8 parity=none
    stop-bits=1 flow-control=none

  1 name=serial1 used-by="" baud-rate=9600 data-bits=8 parity=none stop-bits=1
    flow-control=none

[admin@MikroTik] system serial-console>
```

Using Serial Terminal

Command name: ***/system serial-terminal***

Description

The command is used to communicate with devices and other systems that are connected to router via serial port.

All keyboard input is forwarded to the serial port and all data from the port is output to the connected device. After exiting with [Ctrl]+[Q], the control signals of the port are lowered. The speed and other parameters of serial port may be configured in the **/port** directory of router console. No terminal translation on printed data is performed. It is possible to get the terminal in an unusable state by outputting sequences of inappropriate control characters or random data. Do not connect to devices at an incorrect speed and avoid dumping binary data.

Property Description

port (*name*) - port name to use

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Notes

[Ctrl]+[Q] and [Ctrl]+[X] have special meaning and are used to provide a possibility of exiting from nested serial-terminal sessions:

To send [Ctrl]+[X] to to serial port, press [Ctrl]+[X] [Ctrl]+[X]

To send [Ctrl]+[Q] to to serial port, press [Ctrl]+[X] [Ctrl]+[Q]

Example

To connect to a device connected to the **serial1** port:

```
[admin@MikroTik] system> serial-terminal serial1

[Type Ctrl-Q to return to console]
[Ctrl-X is the prefix key]
```

Console Screen

Home menu level: ***/system console screen***

Description

This facility is created to change line number per screen if you have a monitor connected to router.

Property Description

line-count (*25* | *40* | *50*) - number of lines on monitor

Notes

This parameter is applied only to a monitor, connected to the router.

Example

To set monitor's resolution from 80x25 to 80x40:

```
[admin@MikroTik] system console screen> set line-count=40
[admin@MikroTik] system console screen> print
    line-count: 40
[admin@MikroTik] system console screen>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Software Package Management

Document revision 1.3 (Mon Jul 11 12:42:44 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Related Documents
  Description
Installation (Upgrade)
  Description
  Notes
Uninstalling
  Description
  Property Description
  Notes
  Example
Downgrading
  Description
  Command Description
  Example
System Upgrade
  Description
  Property Description
  Example
Adding Package Source
  Description
  Property Description
  Notes
  Example
Software Package List
  Description

General Information

Summary

The MikroTik RouterOS is distributed in the form of software packages. The basic functionality of the router and the operating system itself is provided by the **system** software package. Other packages contain additional software features as well as support to various network interface cards.

Specifications

License required: ***level1***
Home menu level: ***/system package***
Standards and Technologies: ***FTP***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Hardware usage: ***Not significant***

Related Documents

- ***Basic Setup Guide***
- ***Driver Management***
- ***Software Version Management***
- ***License Management***
- ***Installing RouterOS with NetInstall***
- ***Installing RouterOS with CD-Install***
- ***Installing RouterOS with Floppies***

Description

Features

The modular software package system of MikroTik RouterOS has the following features:

- Ability to extend RouterOS functions by installing additional software packages
- Optimal usage of the storage space by employing modular/compressed system
- Unused software packages can be uninstalled
- The RouterOS functions and the system itself can be easily upgraded
- Multiple packages can be installed at once
- The package dependency is checked before installing a software package. The package will not be installed, if the required software package is missing
- The version of the feature package should be the same as that of the **system** package
- The packages can be uploaded on the router using ftp and installed only when the router is going for shutdown during the reboot process
- If the software package file can be uploaded to the router, then the disk space is sufficient for the installation of the package
- The system can be downgraded to an older version by uploading the needed packages to router via FTP binary mode. After that, execute command **/system package downgrade**

Installation (Upgrade)

Description

Installation or upgrade of the MikroTik RouterOS software packages can be done by uploading the newer version of the software package to the router and rebooting it.

The software package files are compressed binary files, which can be downloaded from the MikroTik's web page download section. The full name of the software package consists of a

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

descriptive name, version number and extension **.npk**, *exempli gratia* **system-2.9beta5.npk**, **routerboard-2.9beta5.npk**.

You should check the available hard disk space prior to downloading the package file by issuing **/system resource print** command. If there is not enough free disk space for storing the upgrade packages, it can be freed up by uninstalling some software packages, which provide functionality not required for your needs. If you have a sufficient amount of free space for storing the upgrade packages, connect to the router using ftp. Use user name and password of a user with full access privileges.

Step-by-Step

- Connect to the router using ftp client
- Select the BINARY mode file transfer
- Upload the software package files to the router
- Check the information about the uploaded software packages using the **/file print** command
- Reboot the router by issuing the **/system reboot** command or by pressing **Ctrl+Alt+Del** keys at the router's console
- After reboot, verify that the packages were installed correctly by issuing **/system package print** command

Notes

The packages uploaded to the router should retain the original name and also be in lowercase.

The installation/upgrade process is shown on the console screen (monitor) attached to the router.

The Free Demo License do not allow software upgrades using ftp. You should do a complete reinstall from floppies, or purchase the license.

Before upgrading the router, please check the current version of the system package and the additional software packages. The versions of additional packages should match the version number of the system software package. The version of the MikroTik RouterOS system software (and the build number) are shown before the console login prompt. Information about the version numbers and build time of the installed MikroTik RouterOS software packages can be obtained using the **/system package print** command.

Uninstalling

Description

Usually, you do not need to uninstall software packages. However, if you have installed a wrong package, or you need additional free space to install a new one, you have to uninstall some unused packages.

In order to uninstall software package, you have to set **uninstall** property for that package to **yes** and reboot the router.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Property Description

uninstall (yes | no; default: **no**) - If set to yes, schedules the package for uninstallation on next reboot.

Notes

If a package is marked for uninstallation, but it is required for another (dependent) package, then the marked package cannot be uninstalled. You should uninstall the dependent package too. For the list of package dependencies see the 'Software Package List; section below. The system package will not be uninstalled even if marked for uninstallation.

Example

Suppose we need to uninstall **security** package from the router:

```
[admin@MikroTik] system package> print
Flags: I - invalid
  #   NAME                VERSION             BUILD-TIME           UNINSTALL
  0   system              2.9beta5            sep/21/2004 13:27:59 no
  1   ppp                 2.9beta5            sep/21/2004 12:31:52 no
  2   advanced-tools      2.9beta5            sep/21/2004 12:31:42 no
  3   dhcp                2.9beta5            sep/21/2004 12:31:49 no
  4   routing             2.9beta5            sep/21/2004 12:31:55 no
  5   security            2.9beta5            sep/21/2004 12:31:47 no
  6   synchronous         2.9beta5            sep/21/2004 12:32:05 no
  7   wireless            2.9beta5            sep/21/2004 12:32:09 no

[admin@MikroTik] system package> set 5 uninstall=yes
[admin@MikroTik] > .. reboot
```

Downgrading

Command name: ***/system package downgrade***

Description

Downgrade option allows you to downgrade the software via FTP without losing your license key or reinstalling the router.

Step-by-Step

- Connect to the router using ftp client
- Select the BINARY mode file transfer
- Upload the software package files to the router
- Check the information about the uploaded software packages using the **/file print** command
- Execute command **/system package downgrade**. The router will downgrade and reboot.
- After reboot, verify that the packages were installed correctly by issuing **/system package print** command

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Command Description

downgrade - this command asks your confirmation and reboots the router. After reboot the software is downgraded (if all needed packages were uploaded to the router)

Example

To downgrade the RouterOS (assuming that all needed packages are already uploaded):

```
[admin@MikroTik] system package> downgrade
Router will be rebooted. Continue? [y/N]: y
system will reboot shortly
```

System Upgrade

Home menu level: ***/system upgrade***

Description

This submenu gives you the ability to download RouterOS software packages from a remote RouterOS router.

Step-by-Step

- Upload desired RouterOS packages to a router (not the one that you will upgrade)
- Add this router's IP address, user name and password to **/system upgrade upgrade-package-source**
- Refresh available software package list **/system upgrade refresh**
- See available packages, using **/system upgrade print** command
- Download selected or all packages from the remote router, using the **download** or **download-all** command

Property Description

download - download packages from list by specifying their numbers

download-all - download all packages that are needed for the upgrade (packages which are available in '/system package print' list)

name (*read-only: name*) - package name

refresh - updates currently available package list

source (*read-only: IP address*) - source IP address of the router from which the package list entry is retrieved

status (*read-only: available* | *scheduled* | *downloading* | *downloaded* | *installed*) - package status

version (*read-only: text*) - version of the package

Example

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

See the available packages:

```
[admin@MikroTik] system upgrade> print
 # SOURCE          NAME              VERSION      STATUS      COMPLETED
 0 192.168.25.8    advanced-tools    2.9beta5     available
 1 192.168.25.8    dhcp              2.9beta5     available
 2 192.168.25.8    hotspot           2.9beta5     available
 3 192.168.25.8    isdn              2.9beta5     available
 4 192.168.25.8    ntp               2.9beta5     available
 5 192.168.25.8    ppp               2.9beta5     available
 6 192.168.25.8    routerboard       2.9beta5     available
 7 192.168.25.8    routing           2.9beta5     available
 8 192.168.25.8    security          2.9beta5     available
 9 192.168.25.8    synchronous       2.9beta5     available
10 192.168.25.8    system            2.9beta5     available
11 192.168.25.8    telephony         2.9beta5     available
12 192.168.25.8    ups               2.9beta5     available
13 192.168.25.8    web-proxy         2.9beta5     available
14 192.168.25.8    wireless          2.9beta5     available
[admin@MikroTik] system upgrade>
```

To upgrade chosen packages:

```
[admin@MikroTik] system upgrade> download 0,1,2,5,6,7,8,9,10,13,14
[admin@MikroTik] system upgrade> print
 # SOURCE          NAME              VERSION      STATUS      COMPLETED
 0 192.168.25.8    advanced-tools    2.9beta5     downloaded
 1 192.168.25.8    dhcp              2.9beta5     downloading 16 %
 2 192.168.25.8    hotspot           2.9beta5     scheduled
 3 192.168.25.8    isdn              2.9beta5     available
 4 192.168.25.8    ntp               2.9beta5     available
 5 192.168.25.8    ppp               2.9beta5     scheduled
 6 192.168.25.8    routerboard       2.9beta5     scheduled
 7 192.168.25.8    routing           2.9beta5     scheduled
 8 192.168.25.8    security          2.9beta5     scheduled
 9 192.168.25.8    synchronous       2.9beta5     scheduled
10 192.168.25.8    system            2.9beta5     scheduled
11 192.168.25.8    telephony         2.9beta5     available
12 192.168.25.8    ups               2.9beta5     available
13 192.168.25.8    web-proxy         2.9beta5     scheduled
14 192.168.25.8    wireless          2.9beta5     scheduled
[admin@MikroTik] system upgrade>
```

Adding Package Source

Home menu level: ***/system upgrade upgrade-package-source***

Description

In this submenu you can add remote routers from which to download the RouterOS software packages.

Property Description

address (*IP address*) - source IP address of the router from which the package list entry will be retrieved

password (*text*) - password of the remote router

user (*text*) - username of the remote router

Notes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

After specifying a remote router in **/system upgrade upgrade-package-source**, you can type **/system upgrade refresh** to refresh the package list and **/system upgrade print** to see all available packages.

Example

To add a router with IP address **192.168.25.8**, username **admin** and no password:

```
/system upgrade upgrade-package-source add address=192.168.25.8 user=admin
[admin@MikroTik] system upgrade upgrade-package-source> print
# ADDRESS         USER
0 192.168.25.8    admin
[admin@MikroTik] system upgrade upgrade-package-source>
```

Software Package List

Description

System Software Package

The **system** software package provides the basic functionality of the MikroTik RouterOS, namely:

- IP address management, ARP, static IP routing, policy routing, firewall (packet filtering, content filtering, masquerading, and static NAT), traffic shaping (queues), IP traffic accounting, MikroTik Neighbour Discovery, IP Packet Packing, DNS client settings, IP service (servers)
- Ethernet interface support
- IP over IP tunnel interface support
- Ethernet over IP tunnel interface support
- driver management for Ethernet ISA cards
- serial port management
- local user management
- export and import of router configuration scripts
- backup and restore of the router's configuration
- undo and redo of configuration changes
- network diagnostics tools (ping, traceroute, bandwidth tester, traffic monitor)
- bridge support
- system resource management
- package management
- telnet client and server
- local and remote logging facility
- winbox server as well as winbox executable with some plugins

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

After installing the MikroTik RouterOS, a free license should be obtained from MikroTik to enable the basic system functionality.

Additional Software Feature Packages

The table below shows additional software feature packages, extended functionality provided by them, the required prerequisites and additional licenses, if any.

Name	Contents	Prerequisites	Additional License
advanced-tools	email client, pingers, netwatch and other utilities	none	none
arlan	support for DSSS 2.4GHz 2mbps Aironet ISA cards	none	2.4GHz/5GHz Wireless Client
dhcp	DHCP server and client support	none	none
gps	support for GPS devices	none	none
hotspot	HotSpot gateway	none	any additional license
isdn	support for ISDN devices	ppp	none
lcd	support for informational LCD display	none	none
ntp	network time protocol support	none	none
ppp	support for PPP, PPTP, L2TP, PPPoE and ISDN PPP	none	none
radiolan	Provides support for 5.8GHz RadioLAN cards	none	2.4GHz/5GHz Wireless Client
routerboard	support for RouterBoard-specific functions and utilities	none	none
routing	support for RIP, OSPF and BGP4	none	none
security	support for IPSEC, SSH and secure WinBox connections	none	none
synchronous	support for Frame Relay and Moxa C101, Moxa C502,	none	Synchronous

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

	Farsync, Cyclades PC300, LMC SBE and XPeed synchronous cards		
telephony	**IP telephony support (H.323)**	**none**	**none**
thinrouter-pcipc	**forces PCI-to-CardBus Bridge to use IRQ 11 as in ThinRouters**	**none**	**none**
ups	**APC Smart Mode UPS support**	**none**	**none**
web-proxy	**HTTP Web proxy support**	**none**	**none**
wireless	**Provides support for Cisco Aironet cards, PrismII and Atheros wireless stations and APs**	**none**	**2.4GHz/5GHz Wireless Client / 2.4GHz/5GHz Wireless Server (optional)**

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Software Version Management

Document revision 1.3 (Mon Jul 11 12:44:07 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
System Upgrade
  Related Documents
  Description
  Property Description
  Example
Adding Package Source
  Description
  Property Description
  Notes
  Example

General Information

Summary

To upgrade RouterOS to a more recent version, you can simply transfer the packages to router via ftp, using the **binary** transfer mode, and then just rebooting the router.

This manual discusses a more advanced method how to upgrade a router automatically. If you have more than one router then this can be useful.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/system upgrade***
Standards and Technologies: ***None***
Hardware usage: ***Not significant***

System Upgrade

Home menu level: ***/system upgrade***

Related Documents

- ***Software Package Management***
- ***License Management***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

In this submenu you can see available packages and are able to choose which to install from a remote router.

At first you upload new packages to the router via ftp, using the **binary** data transfer mode. Then (from another router, which you will upgrade) add the router's IP on which are the packages listed in the **/system upgrade upgrade-package-source** list. Afterwards, you type **/system upgrade refresh** to update the available package list. To see all available packages, choose **/system upgrade print** command.

Property Description

download - download packages from list by specifying their numbers

download-all - download all packages that are needed for the upgrade (packages which are available in '/system package print' list)

name (*read-only: name*) - package name

refresh - updates currently available package list

source (*read-only: IP address*) - source IP address of the router from which the package list entry is retrieved

status (*read-only: available* | *scheduled* | *downloading* | *downloaded* | *installed*) - package status

version (*read-only: text*) - version of the package

Example

See the available packages:

```
[admin@MikroTik] system upgrade> print
 # SOURCE          NAME             VERSION      STATUS      COMPLETED
 0 192.168.25.8    advanced-tools   2.9          available
 1 192.168.25.8    dhcp             2.9          available
 2 192.168.25.8    hotspot          2.9          available
 3 192.168.25.8    isdn             2.9          available
 4 192.168.25.8    ntp              2.9          available
 5 192.168.25.8    ppp              2.9          available
 6 192.168.25.8    routerboard      2.9          available
 7 192.168.25.8    routing          2.9          available
 8 192.168.25.8    security         2.9          available
 9 192.168.25.8    synchronous      2.9          available
10 192.168.25.8    system           2.9          available
11 192.168.25.8    telephony        2.9          available
12 192.168.25.8    ups              2.9          available
13 192.168.25.8    web-proxy        2.9          available
14 192.168.25.8    wireless         2.9          available
[admin@MikroTik] system upgrade>
```

To upgrade chosen packages:

```
[admin@MikroTik] system upgrade> download 0,1,2,5,6,7,8,9,10,13,14
[admin@MikroTik] system upgrade> print
 # SOURCE          NAME             VERSION      STATUS      COMPLETED
 0 192.168.25.8    advanced-tools   2.9          downloaded
 1 192.168.25.8    dhcp             2.9          downloading 16 %
 2 192.168.25.8    hotspot          2.9          scheduled
 3 192.168.25.8    isdn             2.9          available
 4 192.168.25.8    ntp              2.9          available
 5 192.168.25.8    ppp              2.9          scheduled
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  6 192.168.25.8    routerboard      2.9          scheduled
  7 192.168.25.8    routing          2.9          scheduled
  8 192.168.25.8    security         2.9          scheduled
  9 192.168.25.8    synchronous      2.9          scheduled
 10 192.168.25.8    system           2.9          scheduled
 11 192.168.25.8    telephony        2.9          available
 12 192.168.25.8    ups              2.9          available
 13 192.168.25.8    web-proxy        2.9          scheduled
 14 192.168.25.8    wireless         2.9          scheduled
 [admin@MikroTik] system upgrade>
```

Adding Package Source

Home menu level: ***/system upgrade upgrade-package-source***

Description

Here can you specify IP address, username and password of the remote hosts from which you will be able to get packages.

Property Description

address (*IP address*) - source IP address of the router from which the package list entry will be retrieved

password (*text*) - password of the remote router

user (*text*) - username of the remote router

Notes

After specifying a remote router in '/system upgrade upgrade-package-source', you can type '/system upgrade refresh' to refresh the package list and '/system upgrade print' to see all available packages.

Example

To add a router, with username **admin** and no password, from which the packages will be retrieved:

```
[admin@MikroTik] system upgrade upgrade-package-source> print
# ADDRESS         USER
0 192.168.25.8    admin
[admin@MikroTik] system upgrade upgrade-package-source>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

SSH (Secure Shell) Server and Client

Document revision 2.0 (Fri Mar 05 09:09:40 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Additional Documents
SSH Server
  Description
  Property Description
  Example
SSH Client
  Property Description
  Example

General Information

Summary

SSH Client authenticates server and encrypts traffic between the client and server. You can use SSH just the same way as telnet - you run the client, tell it where you want to connect to, give your username and password, and everything is the same after that. After that you won't be able to tell that you're using SSH. The SSH feature can be used with various SSH Telnet clients to securely connect to and administrate the router.

The MikroTik RouterOS supports:

- SSH 1.3, 1.5, and 2.0 protocol standards
- server functions for secure administration of the router
- telnet session termination with 40 bit RSA SSH encryption is supported
- secure ftp is supported
- preshared key authentication is not supported

The MikroTik RouterOS has been tested with the following SSH telnet terminals:

- PuTTY
- Secure CRT
- OpenSSH GNU/Linux client

Specifications

Packages required: ***security***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

License required: ***level1***
Home menu level: ***/system ssh***
Standards and Technologies: ***SSH***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***

Additional Documents

- ***http://www.freessh.org/***

SSH Server

Home menu level: ***/ip service***

Description

SSH Server is already up and running after MikroTik router installation. The default port of the service is 22. You can set a different port number.

Property Description

name (*name*) - service name

port (*integer*: 1..65535) - port the service listens to

address (*IP address* | *netmask*; default: **0.0.0.0/0**) - IP address from which the service is accessible

Example

Let's change the default SSH port (22) to 65 on which the SSH server listens for requests:

```
[admin@MikroTik] ip service> set ssh port=65
[admin@MikroTik] ip service> print
Flags: X - disabled, I - invalid
 #   NAME                                PORT  ADDRESS            CERTIFICATE
 0   telnet                              23    0.0.0.0/0
 1   ftp                                 21    0.0.0.0/0
 2   www                                 80    0.0.0.0/0
 3   ssh                                 65    0.0.0.0/0
 4 X www-ssl                             443   0.0.0.0/0
[admin@MikroTik] ip service>
```

SSH Client

Command name: ***/system ssh***

Property Description

port (*integer*; default: **22**) - which TCP port to use for SSH connection to a remote host

user (*text*; default: **admin**) - username for the SSH login

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

```
[admin@MikroTik] > /system ssh 192.168.0.1 user=pakalns port=22
admin@192.168.0.1's password:

  MMM      MMM       KKK                          TTTTTTTTTTT      KKK
  MMMM    MMMM       KKK                          TTTTTTTTTTT      KKK
  MMM MMMM MMM  III  KKK  KKK  RRRRRR     OOOOOO      TTT     III  KKK  KKK
  MMM  MM  MMM  III  KKKKK     RRR  RRR  OOO  OOO     TTT     III  KKKKK
  MMM      MMM  III  KKK KKK   RRRRRR    OOO  OOO     TTT     III  KKK KKK
  MMM      MMM  III  KKK  KKK  RRR  RRR   OOOOOO      TTT     III  KKK  KKK

  MikroTik RouterOS 2.9rc7 (c) 1999-2005       http://www.mikrotik.com/


Terminal unknown detected, using single line input mode
[admin@MikroTik] >
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Telnet Server and Client

Document revision 2.1 (Mon Jul 19 07:31:04 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
Telnet Server
  Description
  Example
Telnet Client
  Description
  Example

General Information

Summary

MikroTik RouterOS has a build-in Telnet server and client features. These two are used to communicate with other systems over a network.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/system, /ip service***
Standards and Technologies: ***Telnet (RFC 854)***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***System Resource Management***

Telnet Server

Home menu level: ***/ip service***

Description

Telnet protocol is intended to provide a fairly general, bi-directional, eight-bit byte oriented communications facility. The main goal is to allow a standard method of interfacing terminal devices to each other.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

MikroTik RouterOS implements industry standard Telnet server. It uses port 23, which must not be disabled on the router in order to use the feature.

You can enable/disable this service or allow the use of the service to certain IP addresses.

Example

```
[admin@MikroTik] ip service> print detail
Flags: X - disabled, I - invalid
 0   name="telnet" port=23 address=0.0.0.0/0

 1   name="ftp" port=21 address=0.0.0.0/0

 2   name="www" port=80 address=0.0.0.0/0

 3   name="hotspot" port=8088 address=0.0.0.0/0

 4   name="ssh" port=65 address=0.0.0.0/0

 5 X name="hotspot-ssl" port=443 address=0.0.0.0/0 certificate=none
[admin@MikroTik] ip service>
```

Telnet Client

Command name: ***/system telnet [IP address] [port]***

Description

MikroTik RouterOS telnet client is used to connect to other hosts in the network via Telnet protocol.

Example

An example of Telnet connection:

```
[admin@MikroTik] > system telnet 172.16.0.1
Trying 172.16.0.1...
Connected to 172.16.0.1.
Escape character is '^]'.

MikroTik v2.9
Login: admin
Password:

  MMM      MMM       KKK                          TTTTTTTTTTT      KKK
  MMMM    MMMM       KKK                          TTTTTTTTTTT      KKK
  MMM MMMM MMM  III  KKK  KKK  RRRRRR     OOOOOO      TTT     III  KKK  KKK
  MMM  MM  MMM  III  KKKKK     RRR  RRR  OOO  OOO     TTT     III  KKKKK
  MMM      MMM  III  KKK KKK   RRRRRR    OOO  OOO     TTT     III  KKK KKK
  MMM      MMM  III  KKK  KKK  RRR  RRR   OOOOOO      TTT     III  KKK  KKK

  MikroTik RouterOS 2.9 (c) 1999-2004             http://www.mikrotik.com/


Terminal unknown detected, using single line input mode
[admin@MikroTik] >
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Terminal Console

Document revision 1.0 (Mon Nov 8 13:15:54 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
Common Console Functions
  Description
  Example
Lists and Item Names
  Description
  Notes
  Example
Quick Typing
  Description
  Notes
Additional Information
  Description
General Commands
  Description
  Command Description
Safe Mode
  Description

General Information

Summary

The Terminal Console is used for accessing the MikroTik Router's configuration and management features using text terminals, *id est* remote terminal clients or locally attached monitor and keyboard. The Terminal Console is also used for writing scripts. This manual describes the general console operation principles. Please consult the Scripting Manual on some advanced console commands and on how to write scripts.

Specifications

Packages required: ***system***
License required: ***level1***
Hardware usage: ***Not significant***

Related Documents

- ***Scripting Host and Complementary Tools***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Common Console Functions

Description

The console allows configuration of the router's settings using text commands. Although the command structure is similar to the Unix shell, you can get additional information about the command structure in the **Scripting Host and Complementary Tools** manual. Since there is a lot of available commands, they are split into groups organized in a way of hierarchical menu levels. The name of a menu level reflects the configuration information accessible in the relevant section, *exempli gratia* /**ip hotspot**.

In general, all menu levels hold the same commands. The difference is expressed mainly in command parameters.

Example

For example, you can issue the **/ip route print** command:

```
[admin@MikroTik] > /ip route print
Flags: A - active, X - disabled, I - invalid, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf, d - dynamic
 #     DST-ADDRESS        G GATEWAY         DISTANCE   INTERFACE
 0 ADC 1.1.1.0/24                                      isp2
 1 A S 2.2.2.0/24         r 1.1.1.2         0          isp2
 2 ADC 3.3.3.0/24                                      bonding1
 3 ADC 10.1.0.0/24                                     isp1
 4 A S 0.0.0.0/0          r 10.1.0.1        0          isp1

[admin@MikroTik] >
```

Instead of typing **ip route** path before each command, the path can be typed only once to move into this particular branch of menu hierarchy. Thus, the example above could also be executed like this:

```
[admin@MikroTik] > ip route
[admin@MikroTik] ip route> print
Flags: A - active, X - disabled, I - invalid, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf, d - dynamic
 #     DST-ADDRESS        G GATEWAY         DISTANCE   INTERFACE
 0 ADC 1.1.1.0/24                                      isp2
 1 A S 2.2.2.0/24         r 1.1.1.2         0          isp2
 2 ADC 3.3.3.0/24                                      bonding1
 3 ADC 10.1.0.0/24                                     isp1
 4 A S 0.0.0.0/0          r 10.1.0.1        0          isp1

[admin@MikroTik] ip route>
```

Notice that the prompt changes in order to reflect where you are located in the menu hierarchy at the moment . To move to the top level again, type /:

```
[admin@MikroTik] > /ip route
[admin@MikroTik] ip route> /
[admin@MikroTik] >
```

To move up one command level, type **..**:

```
[admin@MikroTik] ip route> ..
[admin@MikroTik] ip>
```

You can also use / and **..** to execute commands from other menu levels without changing the current level:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] ip route> /ping 10.0.0.1
10.0.0.1 ping timeout
2 packets transmitted, 0 packets received, 100% packet loss
[admin@MikroTik] ip firewall nat> .. service-port print
Flags: X - disabled, I - invalid
 #   NAME                                                                 PORTS
 0   ftp                                                                  21
 1   tftp                                                                 69
 2   irc                                                                  6667
 3 X h323
 4   quake3
 5   mms
 6   gre
 7   pptp
[admin@MikroTik] ip firewall nat>
```

Lists and Item Names

Description

Lists

Many of the command levels operate with arrays of items: interfaces, routes, users etc. Such arrays are displayed in similarly looking lists. All items in the list have an item number followed by its parameter values.

To change parameters of an item, you have to specify it's number to the set command.

Item Names

Some lists have items that have specific names assigned to each. Examples are **interface** or **user** levels. There you can use item names instead of item numbers.

You do not have to use the **print** command before accessing items by name. As opposed to numbers, names are not assigned by the console internally, but are one of the items' properties. Thus, they would not change on their own. However, there are all kinds of obscure situations possible when several users are changing router's configuration at the same time. Generally, item names are more "stable" than the numbers, and also more informative, so you should prefer them to numbers when writing console scripts.

Notes

Item numbers are assigned by **print** command and are not constant - it is possible that two successive **print** commands will order items differently. But the results of last **print** commands are memorized and thus, once assigned, item numbers can be used even after **add**, **remove** and **move** operations (after **move** operation item numbers are moved with the items). Item numbers are assigned on per session basis, they will remain the same until you quit the console or until the next **print** command is executed. Also, numbers are assigned separately for every item list, so **ip address print** would not change numbers for **interface** list.

Example

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] interface> set 0 mtu=1200
ERROR: item number must be assigned by a print command
use print command before using an item number in a command
[admin@MikroTik] interface> print
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE    MTU
 0  R Public                      ether           0          0          1500
 1  R Local                       ether           0          0          1500
 2  R wlan1                       wlan            0          0          1500
[admin@MikroTik] interface> set 0
disabled  mtu  name  rx-rate  tx-rate
[admin@MikroTik] interface> set 0 mtu=1200
[admin@MikroTik] interface> set wlan1 mtu=1300
[admin@MikroTik] interface> print
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE    MTU
 0  R Public                      ether           0          0          1200
 1  R Local                       ether           0          0          1500
 2  R wlan1                       wlan            0          0          1300
[admin@MikroTik] interface>
```

Quick Typing

Description

There are two features in the console that help entering commands much quicker and easier - the [Tab] key completions, and abbreviations of command names. Completions work similarly to the **bash** shell in UNIX. If you press the [Tab] key after a part of a word, console tries to find the command within the current context that begins with this word. If there is only one match, it is automatically appended, followed by a space:

`/inte[Tab]_` becomes `/interface _`

If there is more than one match, but they all have a common beginning, which is longer than that what you have typed, then the word is completed to this common part, and no space is appended:

`/interface set e[Tab]_` becomes `/interface set ether_`

If you've typed just the common part, pressing the tab key once has no effect. However, pressing it for the second time shows all possible completions in compact form:

```
[admin@MikroTik] > interface set e[Tab]_
[admin@MikroTik] > interface set ether[Tab]_
[admin@MikroTik] > interface set ether[Tab]_
ether1 ether5
[admin@MikroTik] > interface set ether_
```

The [Tab] key can be used almost in any context where the console might have a clue about possible values - command names, argument names, arguments that have only several possible values (like names of items in some lists or name of protocol in firewall and NAT rules).You cannot complete numbers, IP addresses and similar values.

Another way to press fewer keys while typing is to abbreviate command and argument names. You can type only beginning of command name, and, if it is not ambiguous, console will accept it as a full name. So typing:

```
[admin@MikroTik] > pi 10.1 c 3 si 100
```

equals to:

```
[admin@MikroTik] > ping 10.0.0.1 count 3 size 100
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Notes

Pressing [Tab] key while entering IP address will do a DNS lookup, instead of completion. If what is typed before cursor is a valid IP address, it will be resolved to a DNS name (reverse resolve), otherwise it will be resolved directly (i.e. to an IP address). To use this feature, DNS server must be configured and working. To avoid input lockups any such lookup will timeout after half a second, so you might have to press [Tab] several times, before the name is actually resolved.

It is possible to complete not only beginning, but also any distinctive substring of a name: if there is no exact match, console starts looking for words that have string being completed as first letters of a multiple word name, or that simply contain letters of this string in the same order. If single such word is found, it is completed at cursor position. For example:

```
[admin@MikroTik] > interface x[TAB]_
[admin@MikroTik] > interface export _

[admin@MikroTik] > interface mt[TAB]_
[admin@MikroTik] > interface monitor-traffic _
```

Additional Information

Description

Built-in Help

The console has a built-in help, which can be accessed by typing **?**. General rule is that help shows what you can type in position where the **?** was pressed (similarly to pressing [Tab] key twice, but in verbose form and with explanations).

Internal Item Numbers

You can specify multiple items as targets to some commands. Almost everywhere, where you can write the number of item, you can also write a list of numbers:

```
[admin@MikroTik] > interface print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                TYPE             MTU
  0  R ether1               ether            1500
  1  R ether2               ether            1500
  2  R ether3               ether            1500
  3  R ether4               ether            1500
[admin@MikroTik] > interface set 0,1,2 mtu=1460
[admin@MikroTik] > interface print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                TYPE             MTU
  0  R ether1               ether            1460
  1  R ether2               ether            1460
  2  R ether3               ether            1460
  3  R ether4               ether            1500
[admin@MikroTik] >
```

General Commands

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

There are some commands that are common to nearly all menu levels, namely: **print, set, remove, add, find, get, export, enable, disable, comment, move**. These commands have similar behavior throughout different menu levels.

Command Description

print - shows all information that's accessible from particular command level. Thus, /system clock print shows system date and time, /ip route print shows all routes etc. If there's a list of items in current level and they are not read-only, i.e. you can change/remove them (example of read-only item list is /system history, which shows history of executed actions), then print command also assigns numbers that are used by all commands that operate with items in this list. - applicable only to lists of items. The action is performed with all items in this list in the same order in which they are given. - forces the print command to use tabular output form - forces the print command to use property=value output form - shows the number of items - prints the contents of the specific submenu into a file. This file will be available in the router's ftp - shows the output from the print command for every interval seconds - prints the oid value, which is useful for SNMP - prints the output without paging, to see printed output which does not fit in the screen, use [Shift]+[PgUp] key combination

It is possible to sort print output. Like this:

```
[admin@MikroTik] interface> print type=ether
Flags: X - disabled, D - dynamic, R - running
 #    NAME                         TYPE            RX-RATE    TX-RATE    MTU
 0  R isp1                         ether           0          0          1500
 1  R isp2                         ether           0          0          1500
[admin@MikroTik] interface>
```

set - allows you to change values of general parameters or item parameters. The set command has arguments with names corresponding to values you can change. Use ? or double [Tab] to see list of all arguments. If there is a list of items in this command level, then set has one action argument that accepts the number of item (or list of numbers) you wish to set up. This command does not return anything.

add - this command usually has all the same arguments as set, except the action number argument. It adds a new item with values you have specified, usually to the end of list (in places where order is relevant). There are some values that you have to supply (like the interface for a new route), other values are set to defaults unless you explicitly specify them. - Copies an existing item. It takes default values of new item's properties from another item. If you do not want to make exact copy, you can specify new values for some properties. When copying items that have names, you will usually have to give a new name to a copy - add command returns internal number of item it has added - places a new item before an existing item with specified position. Thus, you do not need to use the move command after adding an item to the list - controls disabled/enabled state of the newly added item(-s) - holds the description of a newly created item

remove - removes item(-s) from a list - contains number(-s) or name(-s) of item(-s) to remove.

move - changes the order of items in list where one is relevant. Item numbers after move command are left in a consistent, but hardly intuitive order, so it's better to resync them by using print after each move command. - first argument. Specifies the item(-s) being moved. - second argument.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Specifies the item before which to place all items being moved (they are placed at the end of the list if the second argument is omitted).

find - The find command has the same arguments as set, and an additional from argument which works like the from argument with the print command. Plus, find command has flag arguments like disabled, invalid that take values yes or no depending on the value of respective flag. To see all flags and their names, look at the top of print command's output. The find command returns internal numbers of all items that have the same values of arguments as specified.

edit - this command is in every place that has set command, it can be used to edit values of properties, exempli gratia: `[admin@MikroTik] ip route> print Flags: A - active, X - disabled, I - invalid, D - dynamic, C - connect, S - static, r - rip, b - bgp, o - ospf, d - dynamic # DST-ADDRESS G GATEWAY DISTANCE INTERFACE 0 ADC 1.1.1.0/24 isp2 1 A S 2.2.2.0/24 r 1.1.1.2 0 isp2 2 ADC 3.3.3.0/24 bonding1 3 ADC 10.1.0.0/24 isp1 4 A S 0.0.0.0/0 r 10.1.0.1 0 isp1 [admin@MikroTik] ip route> edit 1 gateway`

Safe Mode

Description

It is possible to change router configuration in a way that will make it not accessible except from local console. Usually this is done by accident, but there is no way to undo last change when connection to router is already cut. Safe mode can be used to minimize such risk.

Safe mode is entered by pressing **[Ctrl]+[X]**. To quit safe mode, press **[Ctrl]+[X]** again.

```
[admin@MikroTik] ip route>[Ctrl]+[X]
[Safe Mode taken]

[admin@MikroTik] ip route<SAFE>
```

Message **Safe Mode taken** is displayed and prompt changes to reflect that session is now in safe mode. All configuration changes that are made (also from other login sessions), while router is in safe mode, are automatically undone if safe mode session terminates abnormally. You can see all such changes that will be automatically undone tagged with an **F** flag in system history:

```
[admin@MikroTik] ip route>
[Safe Mode taken]

[admin@MikroTik] ip route<SAFE> add
[admin@MikroTik] ip route<SAFE> /system history print
Flags: U - undoable, R - redoable, F - floating-undo
  ACTION                                   BY                POLICY
F route added                              admin             write
```

Now, if telnet connection is cut, then after a while (TCP timeout is **9** minutes) all changes that were made while in safe mode will be undone. Exiting session by **[Ctrl]+[D]**emphasis> also undoes all safe mode changes, while **/quit** does not.

If another user tries to enter safe mode, he's given following message:

```
[admin@MikroTik] >
Hijacking Safe Mode from someone - unroll/release/don't take it [u/r/d]:
```

- **[u]** - undoes all safe mode changes, and puts the current session in safe mode.
- **[d]** - leaves everything as-is.
- **[r]** - keeps all current safe mode changes, and puts current session in a safe mode. Previous

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

owner of safe mode is notified about this:

```
[admin@MikroTik] ip firewall rule input
[Safe mode released by another user]
```

If too many changes are made while in safe mode, and there's no room in history to hold them all (currently history keeps up to **100** most recent actions), then session is automatically put out of the safe mode, no changes are automatically undone. Thus, it is best to change configuration in small steps, while in safe mode. Pressing **[Ctrl]+[X]** twice is an easy way to empty safe mode action list.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Winbox

Document revision 1.0 (Fri Mar 05 07:59:49 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Description
Troubleshooting
  Description

General Information

Summary

The MikroTik RouterOS can be configured remotely, using **Telnet**, **SSH**, **WinBox Console** or **Webbox**. In this manual we will discuss how to use the interactive **WinBox** console.

Description

The Winbox console is used for accessing the MikroTik Router configuration and management features, using graphical user interface (GUI).

All Winbox interface functions are as close as possible to Console functions: all Winbox functions are exactly in the same hierarchy in Terminal Console and vice versa (except functions that are not implemented in Winbox). That is why there are no Winbox sections in the manual.

The Winbox Console plugin loader, the winbox.exe program, can be retrieved from the MikroTik router, the URL is **http://router_address/winbox/winbox.exe** Use any web browser on Windows 95/98/ME/NT4.0/2000/XP or Linux to retrieve the **winbox.exe** executable file from Router. If your router is not specifically configured, you can also type in the web-browser just **http://router_address**

The Winbox plugins are cached on the local disk for each MikroTik RouterOS version. The plugins are not downloaded, if they are in the cache, and the router has not been upgraded since the last time it has been accessed.

Starting the Winbox Console

When connecting to the MikroTik router via http (TCP port 80 by default), the router's Welcome Page is displayed in the web browser:

By clicking on the Winbox link you can start the winbox.exe download. Choose **Open** to start the Winbox loader program (you can also save this program to your local disk, and run it from there)

The **winbox.exe** program opens the Winbox login window.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

where:

- discovers and shows MNDP (MikroTik Neighbor Discovery Protocol) or CDP (Cisco Discovery Protocol) devices.
- logs on to the router by specified IP address (and the port number if you have changed it from the default value of 80) or MAC Address (if the router is in the same subnet), user name, and password.
- saves the current sessions to the list (to run them, just double-click on an item).
- removes selected item from the list.
- removes all items from the list, clears cache on the local disk, imports addresses from wbx file or exports them to wbx file.
- Secure Mode
  provides privacy and data integrity between WinBox and RouterOS by means of TLS (Transport Layer Security) protocol.
- Keep Password
  Saves password as a plain text on a local hard drive.

The Winbox Console of the router:

The Winbox Console uses TCP port 8291. After logging onto the router you can work with the MikroTik router's configuration through the Winbox console and perform the same tasks as using the regular console.

Overview of Common Functions

You can use the menu bar to navigate through the router's configuration menus, open configuration windows. By double clicking on some list items in the windows you can open configuration windows for the specific items, and so on.

There are some hints for using the Winbox Console:

- To open the required window, simply click on the corresponding menu item
- Add a new entry
- Remove an existing entry
- Enable an item
- Disable an item
- Make or edit a comment
- Refresh a window
- Undo an action
- Redo an action
- Logout from the Winbox Console

Troubleshooting

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

- **Can I run WinBox on Linux?**
- Yes, you can run WinBox and connect to RouterOS, using Wine
- **I cannot open the Winbox Console**
  Check the port and address for **www** service in **/ip service print** list. Make sure the address you are connecting from matches the network you've specified in **address** field and that you've specified the correct port in the Winbox loader. The command **/ip service set www port=80 address=0.0.0.0/0** will change these values to the default ones so you will be able to connect specifying just the correct address of the router in the address field of Winbox loader
- The Winbox Console uses TCP port 8291. Make sure you have access to it through the firewall.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

IP Addresses and ARP

Document revision 1.3 (Tue Sep 20 19:02:32 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
IP Addressing
  Description
  Property Description
  Notes
  Example
Address Resolution Protocol
  Description
  Property Description
  Notes
  Example
Proxy-ARP feature
  Description
  Example
Unnumbered Interfaces
  Description
  Example
Troubleshooting
  Description

General Information

Summary

The following Manual discusses IP address management and the Address Resolution Protocol settings. IP addresses serve as identification when communicating with other network devices using the TCP/IP protocol. In turn, communication between devices in one physical network proceeds with the help of Address Resolution Protocol and ARP addresses.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip address, /ip arp***
Standards and Technologies: ***IP***, ***ARP***
Hardware usage: ***Not significant***

Related Documents

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***Software Package Management***

IP Addressing

Home menu level: ***/ip address***

Description

IP addresses serve for a general host identification purposes in IP networks. Typical (IPv4) address consists of four octets. For proper addressing the router also needs the network mask value, *id est* which bits of the complete IP address refer to the address of the host, and which - to the address of the network. The network address value is calculated by binary **AND** operation from network mask and IP address values. It's also possible to specify IP address followed by slash "/" and amount of bits assigned to a network mask.

In most cases, it is enough to specify the address, the netmask, and the interface arguments. The network prefix and the broadcast address are calculated automatically.

It is possible to add multiple IP addresses to an interface or to leave the interface without any addresses assigned to it. Leaving a physical interface without an IP address is not a must when the bridging between interfaces is used. In case of bridging, the IP address can be assigned to any interface in the bridge, but actually the address will belong to the bridge interface. You can use **/ip address print detail** to see to which interface the address belongs to.

MikroTik RouterOS has following types of addresses:

- **Static** - manually assigned to the interface by a user
- **Dynamic** - automatically assigned to the interface by estabilished ppp, ppptp, or pppoe connections

Property Description

actual-interface (*read-only: name*) - only applicable to logical interfaces like bridges or tunnels. Holds the name of the actual hardware interface the logical one is bound to.

address (*IP address*) - IP address

broadcast (*IP address*; default: **255.255.255.255**) - broadcasting IP address, calculated by default from an IP address and a network mask

disabled (yes | no; default: **no**) - specifies whether the address is disabled or not

interface (*name*) - interface name the IP address is assigned to

netmask (*IP address*; default: **0.0.0.0**) - specifies network address part of an IP address

network (*IP address*; default: **0.0.0.0**) - IP address for the network. For point-to-point links it should be the address of the remote end

Notes

You cannot have two different IP addresses from the same network assigned to the router. *Exempli gratia*, the combination of IP address **10.0.0.1/24** on the **ether1** interface and IP address **10.0.0.132/24** on the **ether2** interface is invalid, because both addresses belong to the same network **10.0.0.0/24**. Use addresses from different networks on different interfaces, or enable **proxy-arp** on

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

ether1 or **ether2**.

Example

```
[admin@MikroTik] ip address> add address=10.10.10.1/24 interface=ether2
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   2.2.2.1/24         2.2.2.0         2.2.2.255       ether2
  1   10.5.7.244/24      10.5.7.0        10.5.7.255      ether1
  2   10.10.10.1/24      10.10.10.0      10.10.10.255    ether2

[admin@MikroTik] ip address>
```

Address Resolution Protocol

Home menu level: ***/ip arp***

Description

Even though IP packets are addressed using IP addresses, hardware addresses must be used to actually transport data from one host to another. Address Resolution Protocol is used to map OSI level 3 IP addreses to OSI level 2 MAC addreses. A router has a table of currently used ARP entries. Normally the table is built dynamically, but to increase network security, it can be built statically by means of adding static entries.

Property Description

address (*IP address*) - IP address to be mapped

interface (*name*) - interface name the IP address is assigned to

mac-address (*MAC address*; default: **00:00:00:00:00:00**) - MAC address to be mapped to

Notes

Maximal number of ARP entries is 8192.

If arp feature is turned off on the interface, i.e., **arp=disabled** is used, ARP requests from clients are not answered by the router. Therefore, static arp entry should be added to the clients as well. For example, the router's IP and MAC addresses should be added to the Windows workstations using the **arp** command:

```
C:\> arp -s 10.5.8.254  00-aa-00-62-c6-09
```

If **arp** property is set to **reply-only** on the interface, then router only replies to ARP requests. Neighbour MAC addresses will be resolved using **/ip arp** statically.

Example

```
[admin@MikroTik] ip arp> add address=10.10.10.10 interface=ether2 mac-address=06 \
\... :21:00:56:00:12
[admin@MikroTik] ip arp> print
Flags: X - disabled, I - invalid, H - DHCP, D - dynamic
  #   ADDRESS         MAC-ADDRESS       INTERFACE
  0 D 2.2.2.2         00:30:4F:1B:B3:D9 ether2
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  1 D 10.5.7.242      00:A0:24:9D:52:A4 ether1
  2   10.10.10.10     06:21:00:56:00:12 ether2
[admin@MikroTik] ip arp>
```

If static arp entries are used for network security on an interface, you should set arp to 'reply-only' on that interface. Do it under the relevant **/interface** menu:

```
[admin@MikroTik] ip arp> /interface ethernet set ether2 arp=reply-only
[admin@MikroTik] ip arp> print
Flags: X - disabled, I - invalid, H - DHCP, D - dynamic
  #   ADDRESS         MAC-ADDRESS       INTERFACE
  0 D 10.5.7.242      00:A0:24:9D:52:A4 ether1
  1   10.10.10.10     06:21:00:56:00:12 ether2

[admin@MikroTik] ip arp>
```

Proxy-ARP feature

Description

A router with properly configured proxy ARP feature acts like a transparent ARP proxy between directly connected networks. Consider the following network diagram:

Suppose the host A needs to communicate to host C. To do this, it needs to know host's C MAC address. As shown on the diagram above, host A has /24 network mask. That makes host A to believe that it is directly connected to the whole 192.168.0.0/24 network. When a computer needs to communicate to another one on a directly connected network, it sends a broadcast ARP request. Therefore host A sends a broadcast ARP request for the host C MAC address.

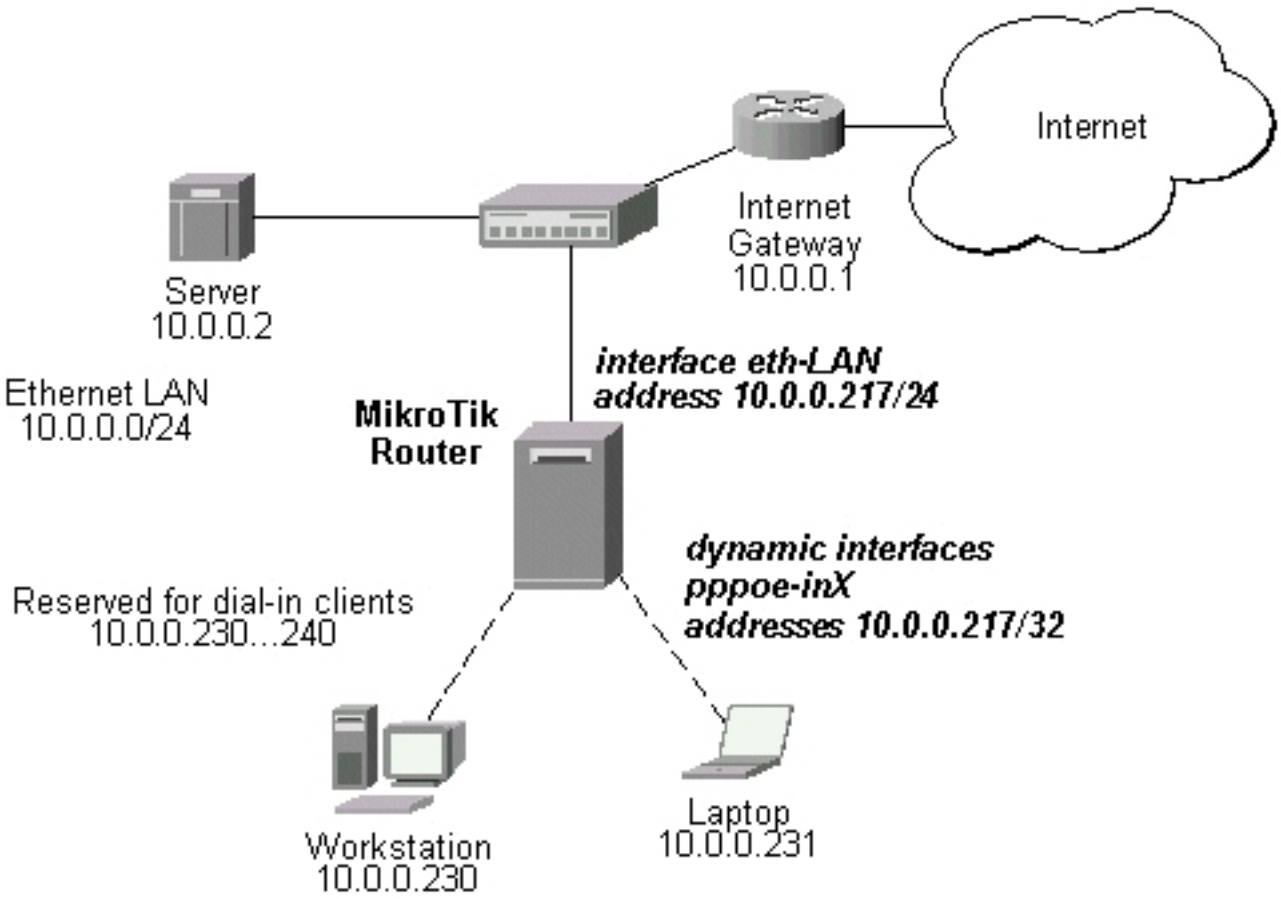
Broadcast ARP requests are sent to the broadcast MAC address FF:FF:FF:FF:FF:FF. Since the ARP request is a broadcast, it will reach all hosts in the network A, including the router R1, but it will not reach host C, because routers do not forward broadcasts by default. A router with enabled proxy ARP knows that the host C is on another subnet and will reply with its own MAC adress. The router with enabled proxy ARP always answer with its own MAC address if it has a route to the destination.

This behaviour can be usefull, for example, if you want to assign dial-in (ppp, pppoe, pptp) clients IP addresses from the same address space as used on the connected LAN.

Example

Consider the following configuration:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



The MikroTik Router setup is as follows:

```
admin@MikroTik] ip arp> /interface ethernet print
Flags: X - disabled, R - running
  #    NAME                MTU   MAC-ADDRESS       ARP
  0  R eth-LAN             1500  00:50:08:00:00:F5 proxy-arp
[admin@MikroTik] ip arp> /interface print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                TYPE              MTU
  0    eth-LAN             ether             1500
  1    prism1              prism             1500
  2 D  pppoe-in25          pppoe-in
  3 D  pppoe-in26          pppoe-in
[admin@MikroTik] ip arp> /ip address print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.217/24      10.0.0.0        10.0.0.255      eth-LAN
  1 D 10.0.0.217/32      10.0.0.230      0.0.0.0         pppoe-in25
  2 D 10.0.0.217/32      10.0.0.231      0.0.0.0         pppoe-in26
[admin@MikroTik] ip arp> /ip route print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0  S 0.0.0.0/0          r 10.0.0.1        1        eth-LAN
    1 DC 10.0.0.0/24        r 0.0.0.0         0        eth-LAN
    2 DC 10.0.0.230/32      r 0.0.0.0         0        pppoe-in25
    3 DC 10.0.0.231/32      r 0.0.0.0         0        pppoe-in26
[admin@MikroTik] ip arp>
```

Unnumbered Interfaces

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

Unnumbered interfaces can be used on serial point-to-point links, e.g., MOXA or Cyclades interfaces. A private address should be put on the interface with the network being the same as the address on the router on the other side of the p2p link (there may be no IP on that interface, but there is an ip for that router).

Example

```
[admin@MikroTik] ip address> add address=10.0.0.214/32 network=192.168.0.1 \
\... interface=pppsync
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.214/32      192.168.0.1     192.168.0.1     pppsync
[admin@MikroTik] ip address>
[admin@MikroTik] ip address> .. route print detail
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    0  S dst-address=0.0.0.0/0 preferred-source=0.0.0.0 gateway=192.168.0.1
         gateway-state=reachable distance=1 interface=pppsync

    1 DC dst-address=192.168.0.1/32 preferred-source=10.0.0.214
         gateway=0.0.0.0 gateway-state=reachable distance=0 interface=pppsync

[admin@MikroTik] ip address>
```

As you can see, a dynamic connected route has been automatically added to the routes list. If you want the default gateway be the other router of the p2p link, just add a static route for it. It is shown as **0** in the example above.

Troubleshooting

Description

- **Router shows that the IP address is invalid**
  Check whether the interface exists to which the IP address is assigned. Or maybe it is disabled. It is also possible that the system has crashed - reboot the router.
- **Router shows that the ARP entry is invalid**
  Check whether the interface exists to which the ARP entry is assigned. Or maybe it is disabled. Check also for an IP address for the particular interface.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

OSPF

Document revision 1.3 (Mon Sep 06 04:56:42 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
General Setup
  Description
  Property Description
  Notes
  Example
Areas
  Description
  Property Description
  Example
Networks
  Description
  Property Description
  Notes
  Example
Interfaces
  Description
  Property Description
  Example
Virtual Links
  Description
  Property Description
  Notes
  Example
Neighbours
  Description
  Property Description
  Notes
  Example
  OSPF backup without using a tunnel
  Routing tables with Revised Link Cost
  Functioning of the Backup

General Information

Summary

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

MikroTik RouterOS implements OSPF Version 2 (RFC 2328). The OSPF protocol is the link-state protocol that takes care of the routes in the dynamic network structure that can employ different paths to its subnetworks. It always chooses shortest path to the subnetwork first.

Specifications

Packages required: ***routing***
License required: ***level3***
Home menu level: ***/routing ospf***
Standards and Technologies: ***OSPF***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***IP Addresses and ARP***
- ***Routes, Equal Cost Multipath Routing, Policy Routing***
- ***Log Management***

Description

Open Shortest Path First protocol is a link-state routing protocol. It's uses a link-state algorithm to build and calculate the shortest path to all known destinations. The shortest path is calculated using the Dijkstra algorithm. OSPF distributes routing information between the routers belonging to a single autonomous system (AS). An AS is a group of routers exchanging routing information via a common routing protocol.

In order to deploy the OSPF all routers it will be running on should be configured in a coordinated manner (note that it also means that the routers should have the same MTU for all the networks advertized by OSPF protocol).

The OSPF protocol is started after you will add a record to the OSPF network list. The routes learned by the OSPF protocol are installed in the routes table list with the distance of 110.

General Setup

Home menu level: ***/routing ospf***

Description

In this section you will learn how to configure basic **OSPF** settings.

Property Description

distribute-default (*never* | *if-installed-as-type-1* | *if-installed-as-type-2* | *always-as-type-1* | *always-as-type-2*; default: **never**) - specifies how to distribute default route. Should be used for ABR (Area Border router) or ASBR (Autonomous System boundary router) settings

- **never** - do not send own default route to other routers

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **if-installed-as-type-1** - send the default route with type 1 metric only if it has been installed (a static default route, or route added by DHCP, PPP, etc.)
- **if-installed-as-type-2** - send the default route with type 2 metric only if it has been installed (a static default route, or route added by DHCP, PPP, etc.)
- **always-as-type-1** - always send the default route with type 1 metric
- **always-as-type-2** - always send the default route with type 2 metric

metric-bgp (*integer*; default: **20**) - specifies the cost of the routes learned from BGP protocol

metric-connected (*integer*; default: **20**) - specifies the cost of the routes to directly connected networks

metric-default (*integer*; default: **1**) - specifies the cost of the default route

metric-rip (*integer*; default: **20**) - specifies the cost of the routes learned from RIP protocol

metric-static (*integer*; default: **20**) - specifies the cost of the static routes

redistribute-bgp (*as-type-1* | *as-type-2* | *no*; default: **no**) - with this setting enabled the router will redistribute the information about all routes learned by the BGP protocol

redistribute-connected (*as-type-1* | *as-type-2* | *no*; default: **no**) - if set, the router will redistribute the information about all connected routes, i.e., routes to directly reachable networks

redistribute-rip (*as-type-1* | *as-type-2* | *no*; default: **no**) - with this setting enabled the router will redistribute the information about all routes learned by the RIP protocol

redistribute-static (*as-type-1* | *as-type-2* | *no*; default: **no**) - if set, the router will redistribute the information about all static routes added to its routing database, i.e., routes that have been created using the /ip route add command

router-id (*IP address*; default: **0.0.0.0**) - OSPF Router ID. If not specified, OSPF uses the largest IP address configured on the interfaces as its router ID

Notes

Within one area, only the router that is connected to another area (i.e. Area border router) or to another AS (i.e. Autonomous System boundary router) should have the propagation of the default route enabled.

OSPF protocol will try to use the shortest path (path with the smallest total cost) if available.

OSPF protocol supports two types of metrics:

- **type1** - external metrics are expressed in the same units as OSPF interface cost. In other words the router expects the cost of a link to a network which is external to AS to be the same order of magnitude as the cost of the internal links.
- **type2** - external metrics are an order of magnitude larger; any type2 metric is considered greater than the cost of any path internal to the AS. Use of type2 external metric assumes that routing between AS is the major cost of routing a packet, and climinates the need conversion of external costs to internal link state metrics.

Both Type 1 and Type 2 external metrics can be used in the AS at the same time. In that event, Type 1 external metrics always take precedence.

In **/ip route** you can see routes with **Io** status. Because router receives routers from itself.

The metric cost can be calculated from line speed by using the formula 10e+8/line speed. The table

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

contains some examples:

network type	cost
ethernet	**10**
T1	**64**
64kb/s	**1562**

Example

To enable the OSPF protocol redisrtibute routes to the connected networks as **type1** metrics with the cost of **1**, you need do the following:

```
[admin@MikroTik] routing ospf> set redistribute-connected=as-type-1 \
\... metric-connected=1
[admin@MikroTik] routing ospf> print
                  router-id: 0.0.0.0
         distribute-default: never
    redistribute-connected: as-type-1
       redistribute-static: no
          redistribute-rip: no
          redistribute-bgp: no
            metric-default: 1
          metric-connected: 1
             metric-static: 20
                metric-rip: 20
                metric-bgp: 20
[admin@MikroTik] routing ospf>
```

Areas

Home menu level: ***/routing ospf area***

Description

OSPF allows collections of routers to be grouped together. Such group is called an area. Each area runs a separate copy of the basic link-state routing algorithm. This means that each area has its own link-state database and corresponding graph

The structure of an area is invisible from the outside of the area. This isolation of knowledge enables the protocol to effect a marked reduction in routing traffic as compared to treating the entire Autonomous System as a single link-state domain

60-80 routers have to be the maximum in one area

Property Description

area-id (*IP address*; default: **0.0.0.0**) - OSPF area identifier. Default area-id=0.0.0.0 is the backbone area. The OSPF backbone always contains all area border routers. The backbone is responsible for distributing routing information between non-backbone areas. The backbone must be contiguous. However, areas do not need to be physical connected to backbone. It can be done with virtual link. The name and area-id for this area can not be changed

authetication (*none* | *simple* | *md5*; default: **none**) - specifies authentication method for OSPF protocol messages

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **none** - do not use authentication
- **simple** - plain text authentication
- **md5** - keyed Message Digest 5 authentication

default-cost (*integer*; default: **1**) - specifies the default cost used for stub areas. Applicable only to area boundary routers

name (*name*; default: **""**) - OSPF area's name

stub (yes | no; default: **no**) - a stub area is an area which is out from part with no routers or areas beyond it. A stub area is configured to avoid AS External Link Advertisements being flooded into the Stub area. One of the reason to configure a Stub area is that the size of the link state database is reduced along with the routing table and less CPU cycles are used to process. Any router which is trying access to a network outside the area sends the packets to the default route

Example

To define additional OSPF area named **local_10** with **area-id=0.0.10.5**, do the following:

```
[admin@WiFi] routing ospf area> add area-id=0.0.10.5 name=local_10
[admin@WiFi] routing ospf area> print
Flags: X - disabled, I - invalid
 #   NAME                    AREA-ID         STUB DEFAULT-COST AUTHENTICATION
 0   backbone                0.0.0.0                           none
 1   local_10                0.0.10.5        no   1            none
[admin@WiFi] routing ospf area>
```

Networks

Home menu level: ***/routing ospf network***

Description

There can be Point-to-Point networks or Multi-Access networks. Multi-Access network can be a broadcast network (a single message can be sent to all routers)

To start the OSPF protocol, you have to define the networks on which it will run and the area ID for each of those networks

Property Description

area (*name*; default: **backbone**) - the OSPF area to be associated with the specified address range

network (*IP address/mask*; default: **20**) - the network associated with the area. The network argument allows defining one or multiple interfaces to be associated with a specific OSPF area. Only directly connected networks of the router may be specified

Notes

You should set the network address exactly the same as the remote point IP address for point-to-point links. The right netmask in this case is **/32**.

Example

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

To enable the OSPF protocol on the 10.10.1.0/24 network, and include it into the backbone area, do the following:

```
[admin@MikroTik] routing ospf network> add area=backbone network=10.10.1.0/24
[admin@MikroTik] routing ospf network> print
Flags: X - disabled
  #   NETWORK            AREA
  0   10.10.1.0/24       backbone
[admin@MikroTik] routing ospf>
```

Interfaces

Home menu level: ***/routing ospf interface***

Description

This facility provides tools for additional in-depth configuration of OSPF interface specific parameters. You do not have to configure interfaces in order to run OSPF

Property Description

authentication-key (*text*; default: **""**) - authentication key have to be used by neighboring routers that are using OSPF's simple password authentication

cost (*integer*: 1..65535; default: **1**) - interface cost expressed as link state metric

dead-interval (*time*; default: **40s**) - specifies the interval after which a neighbor is declared as dead. The interval is advertised in the router's hello packets. This value must be the same for all routers and access servers on a specific network

hello-interval (*time*; default: **10s**) - the interval between hello packets that the router sends on the interface. The smaller the hello-interval, the faster topological changes will be detected, but more routing traffic will ensue. This value must be the same on each end of the adjancency otherwise the adjacency will not form

interface (*name*; default: **all**) - interface on which OSPF will run

- **all** - is used for the interfaces not having any specific settings

priority (*integer*: 0..255; default: **1**) - router's priority. It helps to determine the designated router for the network. When two routers attached to a network both attempt to become the designated router, the one with the higher router's priority takes precedence

retransmit-interval (*time*; default: **5s**) - time between retransmitting lost link state advertisements. When a router sends a link state advertisement (LSA) to its neighbor, it keeps the LSA until it receives back the acknowledgment. If it receives no acknowledgment in time, it will retransmit the LSA. The following settings are recommended: for Broadcast network are 5 seconds and for Point-to-Point network are 10 seconds

transmit-delay (*time*; default: **1s**) - link state transmit delay is the estimated time it takes to transmit a link state update packet on the interface

Example

To add an entry that specifies that **ether2** interface should send Hello packets every 5 seconds, do the following:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] routing ospf> interface add interface=ether2 hello-interval=5s
[admin@MikroTik] routing ospf> interface print
  0 interface=ether2 cost=1 priority=1 authentication-key=""
    retransmit-interval=5s transmit-delay=1s hello-interval=5s
    dead-interval=40s


[admin@MikroTik] routing ospf>
```

Virtual Links

Home menu level: ***/routing ospf virtual-link***

Description

As stated in OSPF RFC, the backbone area must be contiguous. However, it is possible to define areas in such a way that the backbone is no longer contiguous. In this case the system administrator must restore backbone connectivity by configuring virtual links. Virtual link can be configured between two routers through common area called transit area, one of them should have to be connected with backbone. Virtual links belong to the backbone. The protocol treats two routers joined by a virtual link as if they were connected by an unnumbered point-to-point network

Property Description

neighbor-id (*IP address*; default: **0.0.0.0**) - specifies router-id of the neighbour

transit-area (*name*; default: **(unknown)**) - a non-backbone area the two routers have in common

Notes

Virtual links can not be estabilished through stub areas

Example

To add a virtual link with the 10.0.0.201 router through the ex area, do the following:

```
[admin@MikroTik] routing ospf virtual-link> add neighbor-id=10.0.0.201 \
\... transit-area=ex
[admin@MikroTik] routing ospf virtual-link> print
Flags: X - disabled, I - invalid
  #   NEIGHBOR-ID     TRANSIT-AREA
  0   10.0.0.201      ex
[admin@MikroTik] routing ospf virtual-link>
```

Virtual link should be configured on both routers

Neighbours

Home menu level: ***/routing ospf neigbor***

Description

The submenu provides an access to the list of OSPF neighbors, *id est* the routers adjacent to the current router, and supplies brief statistics

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Property Description

address (*read-only: IP address*) - appropriate IP address of the neighbour

backup-dr-id (*read-only: IP address*) - backup designated router's router id for this neighbor

db-summaries (*read-only: integer*) - number of records in link-state database advertised by the neighbour

dr-id (*read-only: IP address*) - designated router's router id for this neighbor

ls-requests (*read-only: integer*) - number of link-state requests

ls-retransmits (*read-only: integer*) - number of link-state retransmits

priority (*read-only: integer*) - the priority of the neigbour which is used in designated router elections via Hello protocol on this network

router-id (*read-only: IP address*) - the router-id parameter of the neighbour

state (*read-only: Down* | *Attempt* | *Init* | *2-Way* | *ExStart* | *Exchange* | *Loading* | *Full*) - the state of the connection:

- **Down** - the connection is down
- **Attempt** - the router is sending Hello protocol packets
- **Init** - Hello packets are exchanged between routers to create a Neighbour Relationship
- **2-Way** - the routers add each other to their Neighbour database and they become neighbours
- **ExStart** - the DR (Designated Router) and BDR (Backup Designated Router) create an adjancency with each other and they begin creating their link-state databases using Database Description Packets
- **Exchange** - is the process of discovering routes by exchanging Database Description Packets
- **Loading** - receiving information from the neighbour
- **Full** - the link-state databases are completely synchronized. The routers are routing traffic and continue sending each other hello packets to maintain the adjacency and the routing information

state-changes (*read-only: integer*) - number of connection state changes

Notes

The neighbour's list also displays the router itself with 2-Way state

Example

The following text can be observed just after adding an OSPF network:

```
admin@MikroTik] routing ospf> neighbor print
 router-id=10.0.0.204 address=10.0.0.204 priority=1 state="2-Way"
    state-changes=0 ls-retransmits=0 ls-requests=0 db-summaries=0
    dr-id=0.0.0.0 backup-dr-id=0.0.0.0

[admin@MikroTik] routing ospf>
```

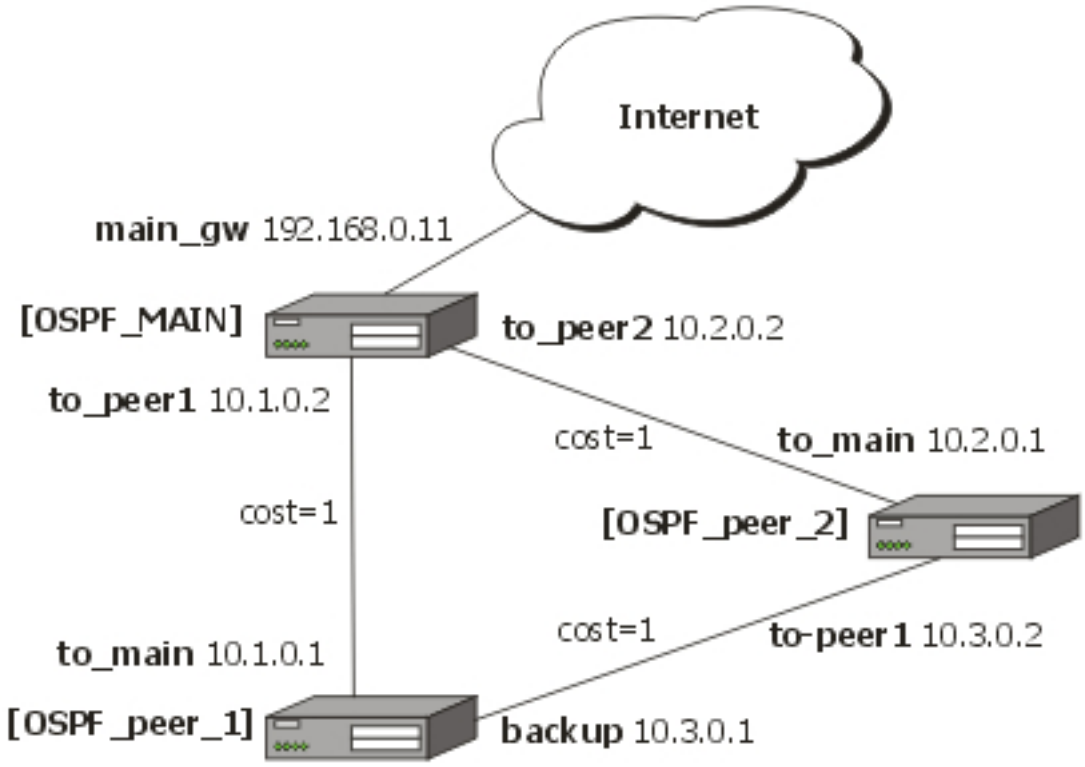
General Information

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

OSPF backup without using a tunnel

Let us assume that the link between the routers OSPF-Main and OSPF-peer-1 is the main one. If it goes down, we want the traffic switch over to the link going through the router OSPF-peer-2.

This example shows how to use OSPF for backup purposes, if you are controlling all the involved routers, and you can run OSPF on them



For this:

1. We introduce an OSPF area with area ID=0.0.0.1, which includes all three routers shown on the diagram
2. Only the OSPF-Main router will have the default route configured. Its interfaces peer1 and peer2 will be configured for the OSPF protocol. The interface main_gw will not be used for distributing the OSPF routing information
3. The routers OSPF-peer-1 and OSPF-peer-2 will distribute their connected route information, and receive the default route using the OSPF protocol

Now let's setup the **OSPF_MAIN** router.

The router should have 3 NICs:

```
[admin@OSPF_MAIN] interface> print
Flags: X - disabled, D - dynamic, R - running
  #      NAME                                    TYPE                  RX-RATE
TX-RATE      MTU
  0   R main_gw                                  ether                   0
0          1500
  1   R to_peer_1                                ether                   0
0          1500
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  2   R to_peer_2                                 ether                    0
0          1500
```

Add all needed ip addresses to interfaces as it is shown here:

```
[admin@OSPF_MAIN] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #    ADDRESS                 NETWORK            BROADCAST          INTERFACE
  0    192.168.0.11/24        192.168.0.0        192.168.0.255      main_gw
  1    10.1.0.2/24            10.1.0.0           10.1.0.255         to_peer_1
  2    10.2.0.2/24            10.2.0.0           10.2.0.255         to_peer_2
```

You should set distribute-default as if-installed-as-type-2, redistribute-connected as as-type-1 and redistribute-static as as-type-2. Metric-connected, metric-static, metric-rip, metric-bgp should be zero

```
[admin@OSPF_MAIN] routing ospf> print
                        router-id: 0.0.0.0
                 distribute-default: if-installed-as-type-2
              redistribute-connected: as-type-1
                 redistribute-static: as-type-2
                    redistribute-rip: no
                    redistribute-bgp: no
                      metric-default: 1
                    metric-connected: 0
                       metric-static: 0
                          metric-rip: 0
                          metric-bgp: 0
```

Define new OSPF area named local_10 with area-id 0.0.0.1:

```
[admin@OSPF_MAIN] routing ospf area> print
Flags: X - disabled, I - invalid
  #    NAME                                AREA-ID             STUB DEFAULT-COST
AUTHENTICATION
  0    backbone                            0.0.0.0
none
  1    local_10                            0.0.0.1              no         1
none
```

Add connected networks with area local_10 in ospf network:

```
[admin@OSPF_MAIN] routing ospf network> print
Flags: X - disabled, I - invalid
  #    NETWORK                 AREA
  0    10.1.0.0/24            local_10
  1    10.2.0.0/24            local_10
```

For main router the configuration is done. Next, you should configure **OSPF_peer_1** router

Enable followong interfaces on **OSPF_peer_1**:

```
[admin@OSPF_peer_1] interface> print
Flags: X - disabled, D - dynamic, R - running
  #      NAME                                TYPE                     RX-RATE
TX-RATE      MTU
  0   R backup                               ether                      0
0         1500
  1   R to_main                              ether                      0
0         1500
```

Assign IP addresses to these interfaces:

```
[admin@OSPF_peer_1] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #    ADDRESS                 NETWORK             BROADCAST          INTERFACE
  0    10.1.0.1/24             10.1.0.0            10.1.0.255         to_main
  1    10.3.0.1/24             10.3.0.0            10.3.0.255         backup
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Set redistribute-connected as as-type-1. Metric-connected, metric-static, metric-rip, metric-bgp should be zero.

```
[admin@OSPF_peer_1] routing ospf> print
                        router-id: 0.0.0.0
                 distribute-default: never
              redistribute-connected: as-type-1
                 redistribute-static: no
                    redistribute-rip: no
                    redistribute-bgp: no
                      metric-default: 1
                    metric-connected: 0
                       metric-static: 0
                          metric-rip: 0
                          metric-bgp: 0
```

Add the same area as in main router:

```
[admin@OSPF_peer_1] routing ospf area> print
Flags: X - disabled, I - invalid
  #    NAME                                AREA-ID             STUB DEFAULT-COST
AUTHENTICATION
  0    backbone                            0.0.0.0
none
  1    local_10                            0.0.0.1              no         1
none
```

Add connected networks with area local_10:

```
[admin@OSPF_peer_1] routing ospf network> print
Flags: X - disabled, I - invalid
  #    NETWORK                  AREA
  0    10.3.0.0/24            local_10
  1    10.1.0.0/24            local_10
```

Finally, set up the **OSPF_peer_2** router. Enable the following interfaces:

```
[admin@OSPF_peer_2] interface> print
Flags: X - disabled, D - dynamic, R - running
  #      NAME                                TYPE                    RX-RATE
TX-RATE      MTU
  0   R to_main                            ether                     0
0         1500
  1   R to_peer_1                          ether                     0
0         1500
```

Add the needed IP addresses:

```
[admin@OSPF_peer_2] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #    ADDRESS                 NETWORK            BROADCAST          INTERFACE
  0    10.2.0.1/24             10.2.0.0           10.2.0.255         to_main
  1    10.3.0.2/24             10.3.0.0           10.3.0.255         to_peer_1
```

Add the same area as in previous routers:

```
[admin@OSPF_peer_2] routing ospf area> print
Flags: X - disabled, I - invalid
  #    NAME                                AREA-ID             STUB DEFAULT-COST
AUTHENTICATION
  0    backbone                            0.0.0.0
none
  1    local_10                            0.0.0.1              no         1
none
```

Add connected networks with the same area:

```
[admin@OSPF_peer_2] routing ospf network> print
Flags: X - disabled, I - invalid
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  #    NETWORK                  AREA
  0    10.2.0.0/24            local_10
  1    10.3.0.0/24            local_10
```

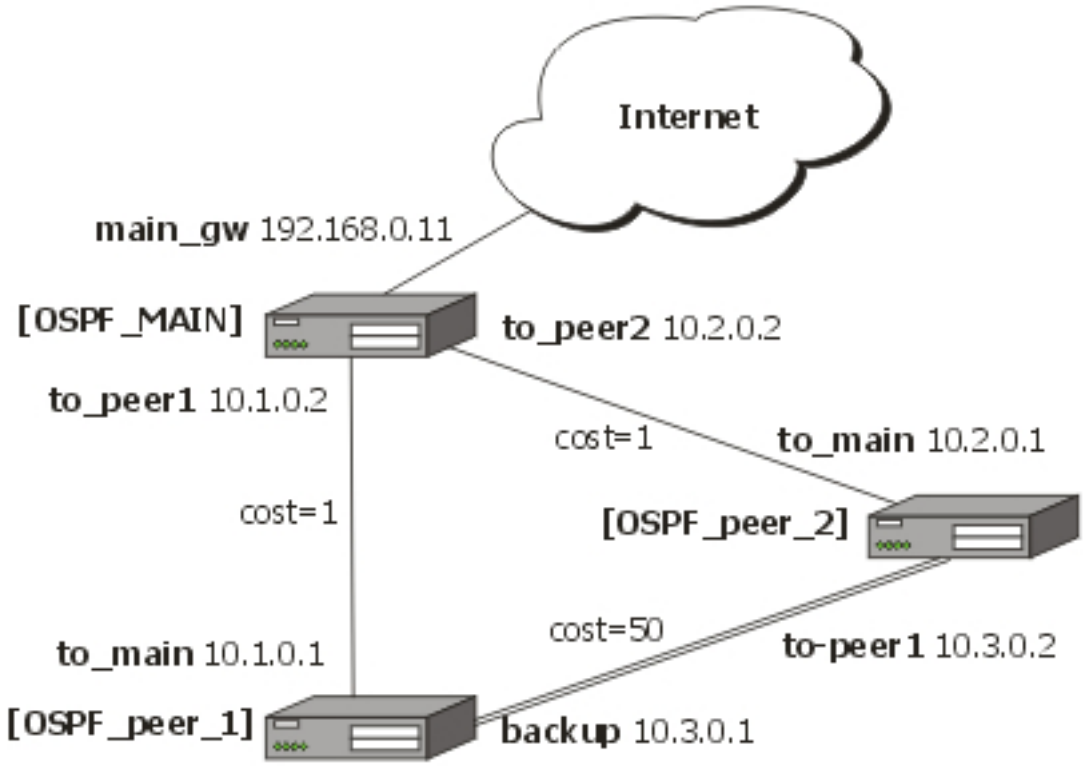
After all routers have been set up as described above, and the links between them are operational, the routing tables of the three routers look as follows:

```
[admin@OSPF_MAIN] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, r - rip, o - ospf, b - bgp
  #      DST-ADDRESS            G GATEWAY             DISTANCE INTERFACE
  0 Io 192.168.0.0/24                                    110
  1 DC 192.168.0.0/24         r 0.0.0.0                  0        main_gw
  2 Do 10.3.0.0/24            r 10.2.0.1                 110      to_peer_2
                              r 10.1.0.1                          to_peer_1
  3 Io 10.2.0.0/24                                  110
  4 DC 10.2.0.0/24            r 0.0.0.0                  0        to_peer_2
  5 Io 10.1.0.0/24                                  110
  6 DC 10.1.0.0/24            r 0.0.0.0                  0        to_peer_1

[admin@OSPF_peer_1] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, r - rip, o - ospf, b - bgp
  #      DST-ADDRESS            G GATEWAY             DISTANCE INTERFACE
  0 Do 192.168.0.0/24         r 10.1.0.2                 110      to_main
  1 Io 10.3.0.0/24                                  110
  2 DC 10.3.0.0/24            r 0.0.0.0                  0        backup
  3 Do 10.2.0.0/24            r 10.1.0.2                 110      to_main
                              r 10.3.0.2                          backup
  4 Io 10.1.0.0/24                                  110
  5 DC 10.1.0.0/24            r 0.0.0.0                  0        to_main

[admin@OSPF_peer_2] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, r - rip, o - ospf, b - bgp
  #      DST-ADDRESS            G GATEWAY             DISTANCE INTERFACE
  0 Do 192.168.0.0/24         r 10.2.0.2                 110      to_main
  1 Io 10.3.0.0/24                                          110
  2 DC 10.3.0.0/24            r 0.0.0.0                  0        to_peer_1
  3 Io 10.2.0.0/24                                  110
  4 DC 10.2.0.0/24            r 0.0.0.0                  0        to_main
  5 Do 10.1.0.0/24            r 10.3.0.1                 110      to_peer_1
                              r 10.2.0.2                          to_main
```

Routing tables with Revised Link Cost

This example shows how to set up link cost. Let us assume, that the link between the routers **OSPF_peer_1** and **OSPF_peer_2** has a higher cost (might be slower, we have to pay more for the traffic through it, etc.).

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



We should change cost value in both routers: **OSPF_peer_1** and **OSPF_peer_2** to 50. To do this, we need to add a following interface:

```
[admin@OSPF_peer_1] routing ospf interface> add interface=backup cost=50
[admin@OSPF_peer_1] routing ospf interface> print
  0 interface=backup cost=50 priority=1 authentication-key=""
 retransmit-interval=5s transmit-delay=1s hello-interval=10s
dead-interval=40s

[admin@OSPF_peer_2] routing ospf interface> add interface=to_peer_1 cost=50
[admin@OSPF_peer_2] routing ospf interface> print
  0 interface=to_peer_1 cost=50 priority=1 authentication-key=""
 retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s
```

After changing the cost settings, we have only one equal cost multipath route left - to the network 10.3.0.0/24 from **OSPF_MAIN** router.

Routes on **OSPF_MAIN** router:

```
[admin@OSPF_MAIN] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, r - rip, o - ospf, b - bgp
  #      DST-ADDRESS           G GATEWAY             DISTANCE INTERFACE
  0 Io 192.168.0.0/24                                110
  1 DC 192.168.0.0/24         r 0.0.0.0                 0       main_gw
  2 Do 10.3.0.0/24            r 10.2.0.1                110     to_peer_2
                              r 10.1.0.1                        to_peer_1
  3 Io 10.2.0.0/24                                  110
  4 DC 10.2.0.0/24            r 0.0.0.0                 0       to_peer_2
  5 Io 10.1.0.0/24                                  110
  6 DC 10.1.0.0/24            r 0.0.0.0                 0       to_peer_1
```

On **OSPF_peer_1**:

```
[admin@OSPF_peer_1] > ip route pr
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

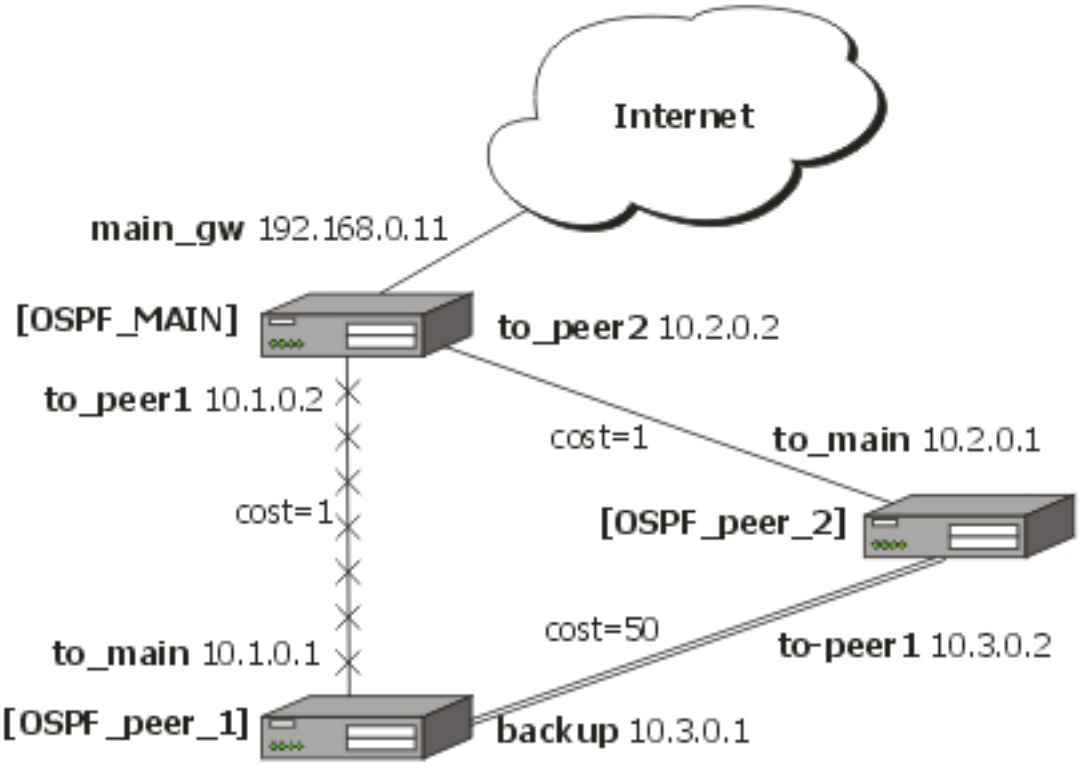
```
C - connect, S - static, r - rip, o - ospf, b - bgp
  #      DST-ADDRESS           G GATEWAY             DISTANCE INTERFACE
  0 Do 192.168.0.0/24        r 10.1.0.2                110       to_main
  1 Io 10.3.0.0/24                                  110
  2 DC 10.3.0.0/24           r 0.0.0.0                 0         backup

  3 Do 10.2.0.0/24           r 10.1.0.2                110       to_main
  4 Io 10.1.0.0/24                                  110
  5 DC 10.1.0.0/24           r 0.0.0.0                 0         to_main
```

On **OSPF_peer_2**:

```
[admin@OSPF_peer_2] > ip route print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, r - rip, o - ospf, b - bgp
  #      DST-ADDRESS           G GATEWAY             DISTANCE INTERFACE
  0 Do 192.168.0.0/24        r 10.2.0.2                110       to_main
  1 Io 10.3.0.0/24                                  110
  2 DC 10.3.0.0/24           r 0.0.0.0                 0         to_peer_1
  3 Io 10.2.0.0/24                                  110
  4 DC 10.2.0.0/24           r 0.0.0.0                 0         to_main
  5 Do 10.1.0.0/24           r 10.2.0.2                110       to_main
```

Functioning of the Backup

If the link between routers **OSPF_MAIN** and **OSPF_peer_1** goes down, we have the following situation:



The OSPF routing changes as follows:

Routes on **OSPF_MAIN** router:

```
[admin@OSPF_MAIN] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, r - rip, o - ospf, b - bgp
  #      DST-ADDRESS           G GATEWAY             DISTANCE INTERFACE
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  0 Io 192.168.0.0/24                                  110
  1 DC 192.168.0.0/24         r 0.0.0.0                  0         main_gw
  2 Do 10.3.0.0/24            r 10.2.0.1                 110       to_peer_2
  3 Io 10.2.0.0/24                                 110
  4 DC 10.2.0.0/24            r 0.0.0.0                  0         to_peer_2
  5 Io 10.1.0.0/24                                 110
  6 DC 10.1.0.0/24            r 0.0.0.0                  0         to_peer_1
```

On **OSPF_peer_1**:

```
[admin@OSPF_peer_1] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, r - rip, o - ospf, b - bgp
  #      DST-ADDRESS           G GATEWAY              DISTANCE INTERFACE
  0 Do 192.168.0.0/24         r 10.3.0.2                 110       backup
  1 Io 192.168.0.0/24                                  110
  2 DC 10.3.0.0/24            r 0.0.0.0                  0         backup
  3 Do 10.2.0.0/24            r 10.3.0.2                 110       backup
  4 Io 10.1.0.0/24                                 110
  5 DC 10.1.0.0/24            r 0.0.0.0                  0         to_main
```

On **OSPF_peer_2**:

```
[admin@OSPF_peer_2] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, r - rip, o - ospf, b - bgp
  #      DST-ADDRESS           G GATEWAY              DISTANCE INTERFACE
  0 Do 192.168.0.0/24         r 10.2.0.2                 110       to_main
  1 Io 10.3.0.0/24                                 110
  2 DC 10.3.0.0/24            r 0.0.0.0                  0         to_peer_1
  3 Io 10.2.0.0/24                                 110
  4 DC 10.2.0.0/24            r 0.0.0.0                  0         to_main
  5 Do 10.1.0.0/24            r 10.2.0.2                 110       to_main
```

The change of the routing takes approximately 40 seconds (the hello-interval setting). If required, this setting can be adjusted, but it should be done on all routers within the OSPF area!

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

RIP

Document revision 1 (Wed Mar 24 12:32:12 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
  Additional Documents
General Setup
  Property Description
  Notes
  Example
Interfaces
  Description
  Property Description
  Notes
  Example
Networks
  Description
  Property Description
  Notes
  Example
Neighbors
  Description
  Property Description
  Example
Routes
  Property Description
  Notes
  Example
  Example

General Information

Summary

MikroTik RouterOS implements RIP Version 1 (RFC1058) and Version 2 (RFC 2453). RIP enables routers in an autonomous system to exchange routing information. It always uses the best path (the path with the fewest number of hops (i.e. routers)) available.

Specifications

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Packages required: ***routing***
License required: ***level3***
Home menu level: ***/routing rip***
Standards and Technologies: ***RIPv1***, ***RIPv2***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***IP Addresses and ARP***
- ***Routes, Equal Cost Multipath Routing, Policy Routing***

Description

Routing Information Protocol (RIP) is one protocol in a series of routing protocols based on Bellman-Ford (or distance vector) algorithm. This Interior Gateway Protocol (IGP) lets routers exchange routing information across a single autonomous system in the way of periodic RIP updates. Routers transmit their own RIP updates to neighboring networks and listen to the RIP updates from the routers on those neighboring networks to ensure their routing table reflects the current state of the network and all the best paths are available. Best path considered to be a path with the fewest hop count (*id est* that include fewer routers).

The routes learned by RIP protocol are installed in the route list (**/ip route print**) with the distance of 120.

Additional Documents

- ***RIPv1 Protocol***
- ***RIPv2 Protocol***
- ***Cisco Systems RIP protocol overview***

General Setup

Property Description

redistribute-static (yes | no; default: **no**) - specifies whether to redistribute static routes to neighbour routers or not

redistribute-connected (yes | no; default: **no**) - specifies whether to redistribute connected routes to neighbour routers or not

redistribute-ospf (yes | no; default: **no**) - specifies whether to redistribute routes learned via OSPF protocol to neighbour routers or not

redistribute-bgp (yes | no; default: **no**) - specifies whether to redistribute routes learned via bgp protocol to neighbour routers or not

metric-static (*integer*; default: **1**) - specifies metric (the number of hops) for the static routes

metric-connected (*integer*; default: **1**) - specifies metric (the number of hops) for the connected

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

routes

metric-ospf (*integer*; default: **1**) - specifies metric (the number of hops) for the routes learned via OSPF protocol

metric-bgp (*integer*; default: **1**) - specifies metric (the number of hops) for the routes learned via BGP protocol

update-timer (*time*; default: **30s**) - specifies frequency of RIP updates

timeout-timer (*time*; default: **3m**) - specifies time interval after which the route is considered invalid

garbage-timer (*time*; default: **2m**) - specifies time interval after which the invalid route will be dropped from neighbor router table

Notes

The maximum metric of RIP route is **15**. Metric higher than **15** is considered 'infinity' and routes with such metric are considered unreachable. Thus RIP cannot be used on networks with more than 15 hops between any two routers, and using **redistribute** metrics larger that **1** further reduces this maximum hop count.

Example

To enable RIP protocol to redistribute the routes to the connected networks:

```
[admin@MikroTik] routing rip> set redistribute-connected=yes
[admin@MikroTik] routing rip> print
       redistribute-static: no
    redistribute-connected: yes
         redistribute-ospf: no
          redistribute-bgp: no
             metric-static: 1
          metric-connected: 1
               metric-ospf: 1
                metric-bgp: 1
              update-timer: 30s
             timeout-timer: 3m
             garbage-timer: 2m
[admin@MikroTik] routing rip>
```

Interfaces

Home menu level: ***/routing rip interface***

Description

In general you do not have to configure interfaces in order to run RIP. This command level is provided only for additional configuration of specific RIP interface parameters.

Property Description

interface (*name*; default: **all**) - interface on which RIP runs

- **all** - sets defaults for interfaces not having any specific settings

send (*v1* | *v1-2* | *v2*; default: **v2**) - specifies RIP protocol update versions to distribute

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

receive (*v1* | *v1-2* | *v2*; default: **v2**) - specifies RIP protocol update versions the router will be able to receive

authentication (*none* | *simple* | *md5*; default: **none**) - specifies authentication method to use for RIP messages

- **none** - no authentication performed
- **simple** - plain text authentication
- **md5** - Keyed Message Digest 5 authentication

authentication-key (*text*; default: **""**) - specifies authentication key for RIP messages

prefix-list-in (*name*; default: **""**) - name of the filtering prefix list for received routes

prefix-list-out (*name*; default: **""**) - name of the filtering prefix list for advertised routes

Notes

It is recommended not to use RIP version 1 wherever it is possible due to security issues

Example

To add an entry that specifies that when advertising routes through the **ether1** interface, prefix list **plout** should be applied:

```
[admin@MikroTik] routing rip> interface add interface=ether1 \
\... prefix-list-out=plout
[admin@MikroTik] routing rip> interface print
Flags: I - inactive
  0   interface=ether1 receive=v2 send=v2 authentication=none
      authentication-key="" prefix-list-in=plout prefix-list-out=none


[admin@MikroTik] routing rip>
```

Networks

Home menu level: ***/routing rip network***

Description

To start the RIP protocol, you have to define the networks on which RIP will run.

Property Description

address (*IP address/mask*; default: **0.0.0.0/0**) - specifies the network on which RIP will run. Only directly connected networks of the router may be specified

netmask (*IP address*; default: **0.0.0.0**) - specifies the network part of the address (if it is not specified in the address argument)

Notes

For point-to-point links you should specify the remote endpoint IP address as the network IP address. For this case the correct **netmask** will be **/32**.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

To enable RIP protocol on **10.10.1.0/24** network:

```
[admin@MikroTik] routing rip network> add address=10.10.1.0/24
[admin@MikroTik] routing rip network> print
  # ADDRESS
  0 10.10.1.0/24
[admin@MikroTik] routing rip>
```

Neighbors

Description

This submenu is used to define a neighboring routers to exchange routing information with. Normally there is no need to add the neighbors, if multicasting is working properly within the network. If there are problems with exchanging routing information, neighbor routers can be added to the list. It will force the router to exchange the routing information with the neighbor using regular unicast packets.

Property Description

address (*IP address*; default: **0.0.0.0**) - IP address of neighboring router

Example

To force RIP protocol to exchange routing information with the **10.0.0.1** router:

```
[admin@MikroTik] routing rip> neighbor add address=10.0.0.1
[admin@MikroTik] routing rip> neighbor print
Flags: I - inactive
  #   ADDRESS
  0   10.0.0.1
[admin@MikroTik] routing rip>
```

Routes

Home menu level: ***/routing rip route***

Property Description

dst-address (*read-only: IP address/mask*) - network address and netmask of destination

gateway (*read-only: IP address*) - last gateway on the route to destination

metric (*read-only: integer*) - distance vector length to the destination network

from (*IP address*) - specifies the IP address of the router from which the route was received

Notes

This list shows routes learned by all dynamic routing protocols (RIP, OSPF and BGP)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

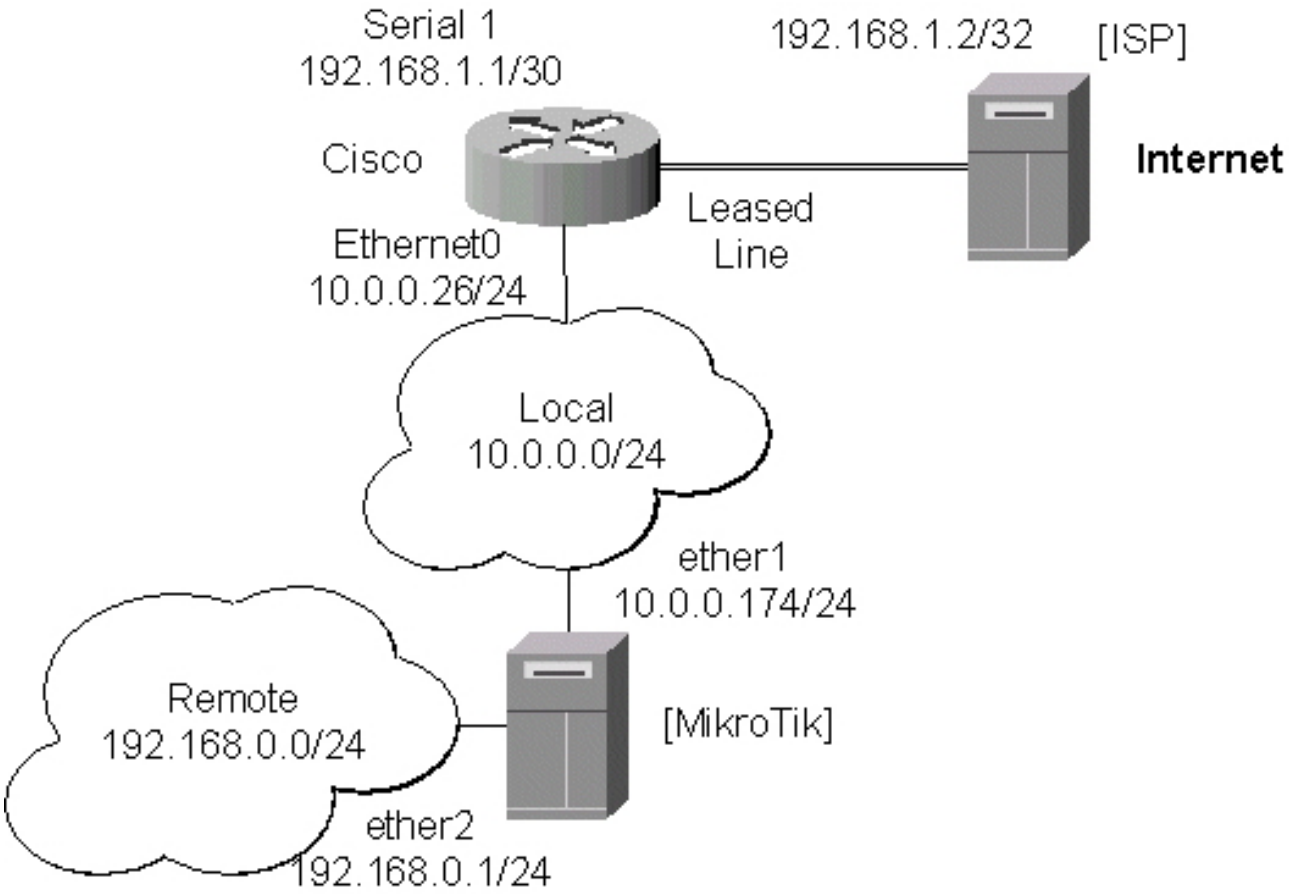
To view the list of the routes:

```
[admin@MikroTik] routing rip route> print
Flags: S - static, R - rip, O - ospf, C - connect, B - bgp
  0 O dst-address=0.0.0.0/32 gateway=10.7.1.254 metric=1 from=0.0.0.0

...

 33 R dst-address=159.148.10.104/29 gateway=10.6.1.1 metric=2 from=10.6.1.1

 34 R dst-address=159.148.10.112/28 gateway=10.6.1.1 metric=2 from=10.6.1.1

[admin@MikroTik] routing rip route>
```

General Information

Example

Let us consider an example of routing information exchange between MikroTik router, a Cisco router and the ISP (also MikroTik) routers:



- MikroTik Router Configuration

```
[admin@MikroTik] > interface print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                TYPE            MTU
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  0  R ether1               ether            1500
  1  R ether2               ether            1500
[admin@MikroTik] > ip address print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.174/24      10.0.0.174      10.0.0.255      ether1
  1   192.168.0.1/24     192.168.0.0     192.168.0.255   ether2
[admin@MikroTik] > ip route print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0 DC 192.168.0.0/24     r 0.0.0.0         0        ether2
    1 DC 10.0.0.0/24        r 0.0.0.0         0        ether1
[admin@MikroTik] >
```

Note, that no default route has been configured. The route will be obtained using the RIP. The necessary configuration of the RIP general settings is as follows:

```
[admin@MikroTik] routing rip> set redistribute-connected=yes
[admin@MikroTik] routing rip> print
       redistribute-static: no
    redistribute-connected: yes
         redistribute-ospf: no
          redistribute-bgp: no
             metric-static: 1
          metric-connected: 1
               metric-ospf: 1
                metric-bgp: 1
              update-timer: 30s
             timeout-timer: 3m
             garbage-timer: 2m

[admin@MikroTik] routing rip>
```

The minimum required configuration of RIP interface is just enabling the network associated with the ether1 interface:

```
[admin@MikroTik] routing rip network> add address=10.0.0.0/2
[admin@MikroTik] routing rip network> print
  # ADDRESS
  0 10.0.0.0/24

[admin@MikroTik] routing rip network>
```

Note, that there is no need to run RIP on the ether2, as no propagation of RIP information is required into the Remote network in this example. The routes obtained by RIP can be viewed in the /routing rip route menu:

```
[admin@MikroTik] routing rip> route print
Flags: S - static, R - rip, O - ospf, C - connect, B - bgp
  0 R dst-address=0.0.0.0/0 gateway=10.0.0.26 metric=2 from=10.0.0.26

  1 C dst-address=10.0.0.0/24 gateway=0.0.0.0 metric=1 from=0.0.0.0

  2 C dst-address=192.168.0.0/24 gateway=0.0.0.0 metric=1 from=0.0.0.0

  3 R dst-address=192.168.1.0/24 gateway=10.0.0.26 metric=1 from=10.0.0.26

  4 R dst-address=192.168.3.0/24 gateway=10.0.0.26 metric=1 from=10.0.0.26

[admin@MikroTik] routing rip>
```

The regular routing table is:

```
[MikroTik] routing rip> /ip route print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0  R 0.0.0.0/0          r 10.0.0.26       120      ether1
    1  R 192.168.3.0/24     r 10.0.0.26       120      ether1
    2  R 192.168.1.0/24     r 10.0.0.26       120      ether1
    3 DC 192.168.0.0/24     r 0.0.0.0         0        ether2
    4 DC 10.0.0.0/24        r 0.0.0.0         0        ether1
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] routing rip>
```

- Cisco Router Configuration

```
Cisco#show running-config
...
interface Ethernet0
 ip address 10.0.0.26 255.255.255.0
 no ip directed-broadcast
!
interface Serial1
 ip address 192.168.1.1 255.255.255.252
 ip directed-broadcast
!
router rip
 version 2
 redistribute connected
 redistribute static
 network 10.0.0.0
 network 192.168.1.0
!
ip classless
!
...
```

The routing table of the Cisco router is:

```
Cisco#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
       U - per-user static route, o - ODR

Gateway of last resort is 192.168.1.2 to network 0.0.0.0

     10.0.0.0/24 is subnetted, 1 subnets
C       10.0.0.0 is directly connected, Ethernet0
R    192.168.0.0/24 [120/1] via 10.0.0.174, 00:00:19, Ethernet0
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.0 is directly connected, Serial1
R    192.168.3.0/24 [120/1] via 192.168.1.2, 00:00:05, Serial1
R*   0.0.0.0/0 [120/1] via 192.168.1.2, 00:00:05, Serial1
Cisco#
```

As we can see, the Cisco router has learned RIP routes both from the MikroTik router (192.168.0.0/24), and from the ISP router (0.0.0.0/0 and 192.168.3.0/24).

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Routes, Equal Cost Multipath Routing, Policy Routing

Document revision 2.2 (Thu Jun 30 10:44:50 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
Routes
  Description
  Property Description
  Notes
  Example
  Static Equal Cost Multi-Path routing
  Standard Policy-Based Routing with Failover

General Information

Summary

The following manual surveys the IP routes management, equal-cost multi-path (ECMP) routing technique, and policy-based routing.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip route***
Standards and Technologies: ***IP (RFC 791)***
Hardware usage: ***Not significant***

Related Documents

- ***IP Addresses and ARP***
- ***Filter***
- ***NAT***

Description

MikroTik RouterOS has following types of routes:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **dynamic routes** - automatically created routes for networks, which are directly accessed through an interface. They appear automatically, when adding a new IP address. Dynamic routes are also added by routing protocols.
- **static routes** - user-defined routes that specify the router which can forward traffic to the specified destination network. They are useful for specifying the default gateway

ECMP (Equal Cost Multi-Path) Routing

This routing mechanism enables packet routing along multiple paths with equal cost and ensures load balancing. With ECMP routing, you can use more than one gateway for one destination network (Note! This approach does not provide failover). With ECMP, a router potentially has several available next hops towards a given destination. A new gateway is chosen for each new source/destination IP pair. It means that, for example, one FTP connection will use only one link, but new connection to a different server will use another link. ECMP routing has another good feature - single connection packets do not get reordered and therefore do not kill TCP performance.

The ECMP routes can be created by routing protocols (RIP or OSPF), or by adding a static route with multiple gateways, separated by a comma (e.g., /ip route add gateway=192.168.0.1,192.168.1.1). The routing protocols may create routes (dynamic) with equal cost automatically, if the cost of the interfaces is adjusted propery. For more information on using routing protocols, please read the corresponding Manual.

Policy-Based Routing

It is a routing approach where the next hop (gateway) for a packet is chosen, based on a policy, which is configured by the network administrator. In RouterOS the procedure the follwing:

- mark the desired packets, with a **routing-mark**
- choose a gateway for the marked packets

Note! In routing process, the router decides which route it will use to send out the packet. Afterwards, when the packet is masqueraded, its source address is taken from the **prefsrc** field.

Routes

Home menu level: ***/ip route***

Description

In this submenu you can configure Static, Equal Cost Multi-Path and Policy-Based Routing and see the routes.

Property Description

as-path (*text*) - manual value of BGP's as-path for outgoing route

atomic-aggregate (yes | no) - BGP attribute. An indication to receiver that it cannot "deaggregate" the prefix

check-gateway (*arp* | *ping*; default: **ping**) - which protocol to use for gateway reachability

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

distance (*integer*: 0..255) - administrative distance of the route. When forwarding a packet, the router will use the route with the lowest administrative distance and reachable gateway

dst-address (*IP address* | *netmask*; default: **0.0.0.0/0**) - destination address and network mask, where netmask is number of bits which indicate network number. Used in static routing to specify the destination which can be reached, using a gateway

- **0.0.0.0/0** - any network

gateway (*IP address*) - gateway host, that can be reached directly through some of the interfaces. You can specify multiple gateways separated by a comma "," for ECMP routes

local-pref (*integer*) - local preference value for a route

med (*integer*) - a BGP attribute, which provides a mechanism for BGP speakers to convey to an adjacent AS the optimal entry point into the local AS

origin (*incomplete* | *igp* | *egp*) - the origin of the route prefix

prefsrc (*IP address*) - source IP address of packets, leaving router via this route

- **0.0.0.0** - prefsrc is determined automatically

prepend (*integer*: 0..16) - number which indicates how many times to prepend AS_NAME to AS_PATH

routing-mark (*name*) - a mark for packets, defined under /ip firewall mangle. Only those packets which have the according routing-mark, will be routed, using this gateway. With this parameter we provide policy based routing

scope (*integer*: 0..255) - a value which is used to recursively lookup the nexthop addresses. Nexthop is looked up only through routes that have scope <= target-scope of the nexthop

target-scope (*integer*: 0..255) - a value which is used to recursively lookup the next-hop addresses. Each nexthop address selects smallest value of target-scope from all routes that use this nexthop address. Nexthop is looked up only through routes that have scope <= target-scope of the nexthop

Notes

You can specify more than one or two gateways in the route. Moreover, you can repeat some routes in the list several times to do a kind of cost setting for gateways.

Example

To add two static routes to networks 10.1.12.0/24 and 0.0.0.0/0 (the default destination address) on a router with two interfaces and two IP addresses:

```
[admin@MikroTik] ip route> add dst-address=10.1.12.0/24 gateway=192.168.0.253
[admin@MikroTik] ip route> add gateway=10.5.8.1
[admin@MikroTik] ip route> print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf
 #     DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
 0 A S 10.1.12.0/24       r 192.168.0.253            Local
 1 ADC 10.5.8.0/24                                   Public
 2 ADC 192.168.0.0/24                                Local
 3 A S 0.0.0.0/0          r 10.5.8.1                 Public
[admin@MikroTik] ip route>
```

General Information

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Static Equal Cost Multi-Path routing

Consider the following situation where we have to route packets from the network **192.168.0.0/24** to 2 gateways - **10.1.0.1** and **10.1.1.1**:

Note that the ISP1 gives us 2Mbps and ISP2 - 4Mbps so we want a traffic ratio 1:2 (1/3 of the source/destination IP pairs from **192.168.0.0/24** goes through ISP1, and 2/3 through ISP2).

IP addresses of the router:

```
[admin@ECMP-Router] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   192.168.0.254/24   192.168.0.0     192.168.0.255   Local
 1   10.1.0.2/28        10.1.0.0        10.1.0.15       Public1
 2   10.1.1.2/28        10.1.1.0        10.1.1.15       Public2
[admin@ECMP-Router] ip address>
```

Add the default routes - one for ISP1 and 2 for ISP2 so we can get the ratio 1:3:

```
[admin@ECMP-Router] ip route> add gateway=10.1.0.1,10.1.1.1,10.1.1.1
[admin@ECMP-Router] ip route> print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf
 #     DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
 0 ADC 10.1.0.0/28                                   Public1
 1 ADC 10.1.1.0/28                                   Public2
 2 ADC 192.168.0.0/24                                Local
 3 A S 0.0.0.0/0          r 10.1.0.1                 Public1
                          r 10.1.1.1                 Public2
                          r 10.1.1.1                 Public2
[admin@ECMP-Router] ip route>
```

Standard Policy-Based Routing with Failover

This example will show how to route packets, using an administrator defined policy. The policy for this setup is the following: route packets from the network **192.168.0.0/24**, using gateway 10.0.0.1, and packets from network **192.168.1.0/24**, using gateway 10.0.0.2. If GW_1 does not respond to pings, use GW_Backup for network 192.168.0.0/24, if GW_2 does not respond to pings, use GW_Backup also for network 192.168.1.0/24 instead of GW_2.

The setup:

Configuration of the IP addresses:

```
[admin@PB-Router] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   192.168.0.1/24     192.168.0.0     192.168.0.255   Local1
 1   192.168.1.1/24     192.168.1.0     192.168.1.255   Local2
 2   10.0.0.7/24        10.0.0.0        10.0.0.255      Public
[admin@PB-Router] ip address>
```

To achieve the described result, follow these configuration steps:

1. Mark packets from network 192.168.0.0/24 with a **new-routing-mark=net1**, and packets from network 192.168.1.0/24 with a **new-routing-mark=net2**:

```
[admin@PB-Router] ip firewall mangle> add src-address=192.168.0.0/24 \
\... action=mark-routing new-routing-mark=net1 chain=prerouting
[admin@PB-Router] ip firewall mangle> add src-address=192.168.1.0/24 \
\... action=mark-routing new-routing-mark=net2 chain=prerouting
[admin@PB-Router] ip firewall mangle> print
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=prerouting src-address=192.168.0.0/24 action=mark-routing
     new-routing-mark=net1

 1   chain=prerouting src-address=192.168.1.0/24 action=mark-routing
     new-routing-mark=net2
[admin@PB-Router] ip firewall mangle>
```

2. Route packets from network 192.168.0.0/24 to gateway GW_1 (10.0.0.2), packets from network 192.168.1.0/24 to gateway GW_2 (10.0.0.3), using the according packet marks. If GW_1 or GW_2 fails (does not reply to pings), route the respective packets to GW_Main (10.0.0.1):

```
[admin@PB-Router] ip route> add gateway=10.0.0.2 routing-mark=net1 \
\... check-gateway=ping
[admin@PB-Router] ip route> add gateway=10.0.0.3 routing-mark=net2 \
\... check-gateway=ping
[admin@PB-Router] ip route> add gateway=10.0.0.1
[admin@PB-Router] ip route> print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf
 #     DST-ADDRESS        PREFSRC         G GATEWAY         DISTANCE INTERFACE
 0 ADC 10.0.0.0/24        10.0.0.7                                   Public
 1 ADC 192.168.0.0/24     192.168.0.1                                Local1
 2 ADC 192.168.1.0/24     192.168.1.1                                Local2
 3 A S 0.0.0.0/0                          r 10.0.0.2                 Public
 4 A S 0.0.0.0/0                          r 10.0.0.3                 Public
 5 A S 0.0.0.0/0                          r 10.0.0.1                 Public
[admin@PB-Router] ip route>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

General Interface Settings

Document revision 1.1 (Fri Mar 05 08:08:52 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Description
Interface Status
  Property Description
  Example
Traffic Monitoring
  Description
  Property Description
  Notes
  Example

General Information

Summary

MikroTik RouterOS supports a variety of Network Interface Cards as well as some virtual interfaces (like Bonding, Bridge, VLAN etc.). Each of them has its own submenu, but there is also a list of all interfaces where some common properties can be configured.

Description

The Manual describes general settings of MikroTik RouterOS interfaces.

Interface Status

Home menu level: ***/interface***

Property Description

name (*text*) - the name of the interface

type (*read-only: arlan* | *bonding* | *bridge* | *cyclades* | *eoip* | *ethernet* | *farsync* | *ipip* | *isdn-client* | *isdn-server* | *l2tp-client* | *l2tp-server* | *moxa-c101* | *moxa-c502* | *mtsync* | *pc* | *ppp-client* | *ppp-server* | *pppoe-client* | *pppoe-server* | *pptp-client* | *pptp-server* | *pvc* | *radiolan* | *sbe* | *vlan* | *wavelan* | *wireless* | *xpeed*) - interface type

mtu (*integer*) - maximum transmission unit for the interface (in bytes)

rx-rate (*integer*; default: **0**) - maximum data rate for receiving data

- **0** - no limits

tx-rate (*integer*; default: **0**) - maximum data rate for transmitting data

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **0** - no limits

Example

To see the list of all available interfaces:

```
[admin@MikroTik] interface> print
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE    MTU
 0  R ether1                      ether           0          0          1500
 1  R bridge1                     bridge          0          0          1500
 2  R ether2                      ether           0          0          1500
 3  R wlan1                       wlan            0          0          1500
[admin@MikroTik] interface>
```

Traffic Monitoring

Command name: ***/interface monitor-traffic***

Description

The traffic passing through any interface can be monitored.

Property Description

received-packets-per-second (*read-only: integer*) - number of packets that interface has received in one second

received-bits-per-second (*read-only: integer*) - number of bits that interface has received in one second

sent-packets-per-second (*read-only: integer*) - number of packets that interface has sent in one second

sent-bits-per-second (*read-only: integer*) - number of bits that interface has sent in one second

Notes

One or more interfaces can be monitored at the same time.

To see overall traffic passing through all interfaces at time, use **aggregate** instead of interface name.

Example

Multiple interface monitoring:

```
/interface monitor-traffic ether1,aggregate
    received-packets-per-second: 9        11
       received-bits-per-second: 4.39kbps 6.19kbps
        sent-packets-per-second: 16       17
           sent-bits-per-second: 101kbps  101kbps
-- [Q quit|D dump|C-z pause]
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

ARLAN 655 Wireless Client Card

Document revision 1.1 (Fri Mar 05 08:12:25 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
Installation
  Example
Wireless Interface Configuration
  Description
  Property Description
  Example
Troubleshooting
  Description

General Information

Summary

The MikroTik RouterOS supports Arlan 655 Wireless Interface client cards. This card fits in the ISA expansion slot and provides transparent wireless communications to other network nodes.

Specifications

Packages required: ***arlan***
License required: ***level4***
Home menu level: ***/interface arlan***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***
- ***Log Management***

Installation

Example

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

To add the driver for Arlan 655 adapter, do the following:

```
[admin@MikroTik]> driver add name=arlan io=0xD000
[admin@MikroTik]> driver print
Flags: I - invalid, D - dynamic
  #   DRIVER                                IRQ IO       MEMORY   ISDN-PROTOCOL
  0 D RealTek 8139
  1   Arlan 655                                 0xD000

[admin@MikroTik] driver>
```

Wireless Interface Configuration

Home menu level: ***/interface arlan***

Description

The wireless card status can be obtained from the two LEDs: the **Status LED** and the **Activity LED**.

Status	Activity	Description
Amber	**Amber**	**ARLAN 655 is functional but nonvolatile memory is not configured**
Blinking Green	**Don't Care**	**ARLAN 655 not registered to an AP (ARLAN mode only)**
Green	**Off**	**Normal idle state**
Green	**Green Flash**	**Normal active state**
Red	**Amber**	**Hardware failure**
Red	**Red**	**Radio failure**

Property Description

name (*name*; default: **arlanN**) - assigned interface name

mtu (*integer*; default: **1500**) - Maximum Transmission Unit

mac-address (*MAC address*) - Media Access Control address

frequency (*2412* | *2427* | *2442* | *2457* | *2465*; default: **2412**) - channel frequency in MHz

bitrate (*1000* | *2000* | *354* | *500*; default: **2000**) - data rate in Kbit/s

sid (*integer*; default: **0x13816788**) - System Identifier. Should be the same for all nodes on the radio network. Must be an even number with maximum length 31 character

add-name (*text*; default: **test**) - card name (optional). Must contain less than 16 characters.

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol setting

tma-mode (*yes* | *no*; default: **no**) - Networking Registration Mode:

- **yes** - ARLAN
- **no** - NON ARLAN

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

```
[admin@MikroTik] > interface print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                                                TYPE            MTU
  0  R outer                                               ether           1500
  1  X arlan1                                              arlan           1500
[admin@MikroTik] interface> enable 1
[admin@MikroTik] > interface print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                                                TYPE            MTU
  0  R outer                                               ether           1500
  1  R arlan1                                              arlan           1500
```

More configuration and statistics parameters can be found under the **/interface arlan** menu:

```
[admin@MikroTik] interface arlan> print
Flags: X - disabled, R - running
  0  R name="arlan1" mtu=1500 mac-address=00:40:96:22:90:C8 arp=enabled
       frequency=2412 bitrate=2000 tma-mode=no card-name="test"
       sid=0x13816788

[admin@MikroTik] interface arlan>
```

You can monitor the status of the wireless interface:

```
[admin@MikroTik] interface arlan> monitor 0
      registered: no
    access-point: 00:00:00:00:00:00
        backbone: 00:00:00:00:00:00

[admin@MikroTik] interface arlan>
```

Suppose we want to configure the wireless interface to accomplish registration on the **AP** with a sid **0x03816788**. To do this, it is enough to change the argument value of **sid** to **0x03816788** and **tma-mode** to **yes**:

```
[admin@MikroTik] interface arlan> set 0 sid=0x03816788 tma-mode=yes
[admin@MikroTik] interface arlan> monitor 0
         registered: yes
    access-point: 00:40:88:23:91:F8
        backbone: 00:40:88:23:91:F9

[admin@MikroTik] interface arlan>
```

Troubleshooting

Description

Keep in mind, that not all combinations of I/O base addresses and IRQs may work on particular motherboard. It is recommended that you choose an IRQ not used in your system, and then try to find an acceptable I/O base address setting. As it has been observed, the IRQ 5 and I/O 0x300 or 0x180 will work in most cases.

- **The driver cannot be loaded because other device uses the requested IRQ.**
  Try to set different IRQ using the DIP switches.
- **The requested I/O base address cannot be used on your motherboard.**
  Try to change the I/O base address using the DIP switches.
- **The pc interface does not show up under the interfaces list**

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Obtain the required license for 2.4/5GHz Wireless Client feature.

- **The wireless card does not register to the Access Point**
  Check the cabling and antenna alignment.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Interface Bonding

Document revision 1.1 (oct-26-2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
- Summary
- Quick Setup Guide
- Specifications
- Related Documents
- Description
- Property Description
- Notes
- Bonding two Eoip tunnels

General Information

Summary

Bonding is a technology that allows to aggregate multiple ethernet-like interfaces into a single virtual link, thus getting higher data rates and providing failover.

Quick Setup Guide

Let us assume that we have 2 NICs in each router (**Router1** and **Router2**) and want to get maximum data rate between 2 routers. To make this possible, follow these steps:

1. Make sure that you do not have IP addresses on interfaces which will be enslaved for bonding interface!
2. Add **bonding** interface on **Router1**:

```
[admin@Router1] interface bonding> add slaves=ether1,ether2
```

   And on **Router2**:

```
[admin@Router2] interface bonding> add slaves=ether1,ether2
```

3. Add addresses to bonding interfaces:

```
[admin@Router1] ip address> add address=172.16.0.1/24 interface=bonding1

[admin@Router2] ip address> add address=172.16.0.2/24 interface=bonding1
```

4. Test the link from **Router1**:

```
[admin@Router1] interface bonding> /pi 172.16.0.2
172.16.0.2 ping timeout
172.16.0.2 ping timeout
172.16.0.2 ping timeout
172.16.0.2 64 byte ping: ttl=64 time=2 ms
172.16.0.2 64 byte ping: ttl=64 time=2 ms
```

   Note that bonding interface needs a couple of seconds to get connectivity with its peer.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/interface bonding***
Standards and Technologies: ***None***
Hardware usage: ***Not significant***

Related Documents

- ***Linux Ethernet Bonding Driver mini-howto***

Description

To provide a proper failover, you should specify **link-monitoring** parameter. It can be:

- MII (Media Independent Interface) type1 or type2 - Media Independent Interface is an abstract layer between the operating system and the NIC which detects whether the link is running (it performs also other functions, but in our case this is the most important).
- ARP - Address Resolution Protocol periodically (for **arp-interval** time) checks the link status.

link-monitoring is used to check whether the link is up or not.

Property Description

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol for the interface

- **disabled** - the interface will not use ARP
- **enabled** - the interface will use ARP
- **proxy-arp** - the interface will use the ARP proxy feature
- **reply-only** - the interface will only reply to the requests originated to its own IP addresses. Neighbour MAC addresses will be resolved using /ip arp statically set table only

arp-interval (*time*; default: **00:00:00.100**) - time in milliseconds which defines how often to monitor ARP requests

arp-ip-targets (*IP address*; default: **""**) - IP target address which will be monitored if link-monitoring is set to arp. You can specify multiple IP addresses, separated by comma

down-delay (*time*; default: **00:00:00**) - if a link failure has been detected, bonding interface is disabled for down-delay time. Value should be a multiple of mii-interval

lacp-rate (*1sec* | *30secs*; default: **30secs**) - Link Aggregation Control Protocol rate specifies how often to exchange with LACPDUs between bonding peer. Used to determine whether link is up or other changes have occured in the network. LACP tries to adapt to these changes providing failover.

link-monitoring (*arp* | *mii-type1* | *mii-type2* | *none*; default: **none**) - method to use for monitoring the link (whether it is up or down)

- **arp** - uses Address Resolution Protocol to determine whether the remote interface is reachable
- **mii-type1** - uses Media Independent Interface type1 to determine link status. Link status

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

determenation relies on the device driver. If bonding shows that the link status is up, when it should not be, then it means that this card don't support this possibility.

- **mii-type2** - uses MII type2 to determine link status (used if mii-type1 is not supported by the NIC)
- **none** - no method for link monitoring is used. If a link fails, it is not considered as down (but no traffic passes through it, thus).

mac-address (*read-only: MAC address*) - MAC address of the bonding interface

mii-interval (*time*; default: **00:00:00.100**) - how often to monitor the link for failures (parameter used only if link-monitoring is mii-type1 or mii-type2)

mtu (*integer*: 68..1500; default: **1500**) - Maximum Transmit Unit in bytes

mode (*802.3ad* | *active-backup* | *balance-alb* | *balance-rr* | *balance-tlb* | *balance-xor* | *broadcast*; default: **balance-rr**) - interface bonding mode. Can be one of:

- **802.3ad** - IEEE 802.3ad dynamic link aggregation. In this mode, the interfaces are aggregated in a group where each slave shares the same speed. If you use a switch between 2 bonding routers, be sure that this switch supports IEEE 802.3ad standard. Provides fault tolerance and load balancing.
- **active-backup** - provides link backup. Only one slave can be active at a time. Another slave becomes active only, if first one fails.
- **balance-alb** - adaptive load balancing. It includes balance-tlb and received traffic is also balanced. Device driver should support for setting the mac address, then it is active. Otherwise balance-alb doesn't work. No special switch is required.
- **balance-rr** - round-robin load balancing. Slaves in bonding interface will transmit and receive data in sequential order. Provides load balancing and fault tolerance.
- **balance-tlb** - Outgoing traffic is distributed according to the current load on each slave. Incoming traffic is received by the current slave. If receiving slave fails, then another slave takes the MAC address of the failed slave. Doesn't require any special switch support.
- **balance-xor** - Use XOR policy for transmit. Provides only failover (in very good quality), but not load balancing, yet.
- **broadcast** - Broadcasts the same data on all interfaces at once. This provides fault tolerance but slows down traffic throughput on some slow machines.

name (*name*) - descriptive name of bonding interface

primary (*name*; default: **none**) - Interface is used as primary output media. If primary interface fails, only then others slaves will be used. This value works only with mode=active-backup

slaves (*name*) - at least two ethernet-like interfaces separated by a comma, which will be used for bonding

up-delay (*time*; default: **00:00:00**) - if a link has been brought up, bonding interface is disabled for up-delay time and after this time it is enabled. Value should be a multiple of mii-interval

Notes

Link failure detection and failover is working significantly better with expensive network cards, for example, made by Intel, then with more cheap ones. For example, on Intel cards failover is taking place in less than a second after link loss, while on some other cards, it may require up to 20 seconds. Also, the Active load balancing (mode=**balance-alb**) does not work on some cheap cards.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

General Information

Bonding two Eoip tunnels

Assume you need to configure the MikroTik router for the following network setup, where you have two offices with 2 ISP for each. You want combine links for getting double speed and provide failover:

We are assuming that connections to Internet through two ISP are configured for both routers.

- Configuration on routers
  - on **Office1**

```
[admin@office1] > /interface print
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE    MTU
 0  R isp1                        ether           0          0  1500
 1  R isp2                        ether           0          0  1500

[admin@office1] > /ip address print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   1.1.1.1/24         1.1.1.0         1.1.1.255       isp2
 1   10.1.0.111/24      10.1.0.0        10.1.0.255      isp1
```

  - on **Office2**

```
[admin@office2] interface> print
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE   MTU
 0  R isp2                        ether           0          0        1500
 1  R isp1                        ether           0          0        1500
[admin@office2] interface> /ip add print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   2.2.2.1/24         2.2.2.0         2.2.2.255       isp2
 1   10.1.0.112/24      10.1.0.0        10.1.0.255      isp1
```

- Eoip tunnel confguration
  - for **Office1** through ISP1

```
[admin@office1] > interface eoip add remote-address=10.1.0.112 tunnel-id=2
\... mac-address=FE:FD:00:00:00:04
[admin@office1] > interface eoip print
Flags: X - disabled, R - running
0  R name="eoip-tunnel2" mtu=1500 mac-address==FE:FD:00:00:00:04 arp=enabled
\... remote-address=10.1.0.112 tunnel-id=2
```

  - for **Office2** through ISP1

```
[admin@office2] > interface eoip add remote-address=10.1.0.111 tunnel-id=2
\... mac-address=FE:FD:00:00:00:02
[admin@office2] > interface eoip print
Flags: X - disabled, R - running
0  R name="eoip-tunnel2" mtu=1500 mac-address=FE:FD:00:00:00:02 arp=enabled
\... remote-address=10.1.0.111 tunnel-id=2
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- for **Office1**through ISP2

```
[admin@office1] > interface eoip add remote-address=2.2.2.1 tunnel-id=1
\... mac-address=FE:FD:00:00:00:03
[admin@office1] interface eoip> print
Flags: X - disabled, R - running
 0  R name="eoip-tunnel1" mtu=1500 mac-address=FE:FD:00:00:00:03 arp=enabled
      remote-address=2.2.2.1 tunnel-id=1

 1  R name="eoip-tunnel2" mtu=1500 mac-address=FE:FD:00:00:00:04 arp=enabled
      remote-address=10.1.0.112 tunnel-id=2
```

- for **Office2**through ISP2

```
[admin@office2] > interface eoip add remote-address=1.1.1.1 tunnel-id=1
\... mac-address=FE:FD:00:00:00:01
[admin@office2] interface eoip> print
Flags: X - disabled, R - running
 0  R name="eoip-tunnel1" mtu=1500 mac-address=FE:FD:00:00:00:01 arp=enabled
      remote-address=1.1.1.1 tunnel-id=1

 1  R name="eoip-tunnel2" mtu=1500 mac-address=FE:FD:00:00:00:02 arp=enabled
      remote-address=10.1.0.111 tunnel-id=2
```

- Bonding confguration
  - for **Office1**

```
[admin@office1] interface bonding> add slaves=eoip-tunnel1,eoip-tunnel2
[admin@office1] interface bonding> print
Flags: X - disabled, R - running
 0  R name="bonding1" mtu=1500 mac-address=00:0C:42:03:20:E7 arp=enabled
      slaves=eoip-tunnel1,eoip-tunnel2 mode=balance-rr primary=none
      link-monitoring=none arp-interval=00:00:00.100 arp-ip-targets=""
      mii-interval=00:00:00.100 down-delay=00:00:00 up-delay=00:00:00
      lacp-rate=30secs
[admin@office1] ip address> add address=3.3.3.1/24 interface=bonding1
[admin@office1] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   1.1.1.1/24         1.1.1.0         1.1.1.255       isp2
 1   10.1.0.111/24      10.1.0.0        10.1.0.255      isp1
 2   3.3.3.1/24         3.3.3.0         3.3.3.255       bonding1
```

  - for **Office2**

```
[admin@office2] interface bonding> add slaves=eoip-tunnel1,eoip-tunnel2
[admin@office2] interface bonding> print
Flags: X - disabled, R - running
 0  R name="bonding1" mtu=1500 mac-address=00:0C:42:03:20:E7 arp=enabled
      slaves=eoip-tunnel1,eoip-tunnel2 mode=balance-rr primary=none
      link-monitoring=none arp-interval=00:00:00.100 arp-ip-targets=""
      mii-interval=00:00:00.100 down-delay=00:00:00 up-delay=00:00:00
      lacp-rate=30secs
[admin@office2] ip address> add address=3.3.3.2/24 interface=bonding1
[admin@office2] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   2.2.2.1/24         2.2.2.0         2.2.2.255       isp2
 1   10.1.0.112/24      10.1.0.0        10.1.0.255      isp1
 2   3.3.3.2/24         3.3.3.0         3.3.3.255       bonding1
[admin@office2] ip address> /ping 3.3.3.1
3.3.3.1 64 byte ping: ttl=64 time=2 ms
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
3.3.3.1 64 byte ping: ttl=64 time=2 ms
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 2/2.0/2 ms
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Bridge

Document revision 2.1 (Fri May 13 12:36:08 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Quick Setup Guide
  Specifications
  Related Documents
  Description
  Additional Documents
Bridge Interface Setup
  Description
  Property Description
  Example
Port Settings
  Description
  Property Description
  Example
Bridge Monitoring
  Description
  Property Description
  Example
Bridge Port Monitoring
  Description
  Property Description
  Example
Bridge Host Monitoring
  Property Description
  Example
Bridge Firewall General Description
  Description
  Property Description
  Notes
Bridge Packet Filter
  Description
  Property Description
Bridge NAT
  Description
  Property Description
Bridge Brouting Facility
  Description
  Property Description
Troubleshooting

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

General Information

Summary

MAC level bridging of Ethernet, Ethernet over IP (EoIP), Prism, Atheros and RadioLAN interfaces are supported. All 802.11a, 802.11b, and 802.11g **client** wireless interfaces (**ad-hoc**, **infrastructure** or **station** mode) do not support this because of the limitations of 802.11. However, it is possible to bridge over the Prism and Atheros based links using the ***WDS*** feature (for Atheros and Prism chipset based cards) or ***Ethernet over IP protocol***.

For preventing loops in a network, you can use the Spanning Tree Protocol (STP). This protocol is also used for configurations with backup links.

Main features:

- Spanning Tree Protocol (STP)
- Multiple bridge interfaces
- Bridge associations on a per-interface basis
- MAC address table can be monitored in real time
- IP address assignment for router access
- Bridge interfaces can be filtered and NATed
- Support for brouting based on bridge packet filter

Quick Setup Guide

To put interface **ether1** and **ether2** in a bridge.

1. Add a bridge interface, called **MyBridge**:

```
/interface bridge add name="MyBridge" disabled=no
```

2. Add **ether1** and **ether2** to **MyBridge** interface:

```
/interface bridge port set ether1,ether2 bridge=MyBridge
```

Specifications

Packages required: ***system***
License required: ***level3***
Home menu level: ***/interface bridge***
Standards and Technologies: ***IEEE801.1D***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***Filter***

Description

Ethernet-like networks (Ethernet, Ethernet over IP, IEEE802.11 in ap-bridge or bridge mode, WDS, VLAN) can be connected together using MAC bridges. The bridge feature allows the interconnection of hosts connected to separate LANs (using EoIP, geographically distributed networks can be bridged as well if any kind of IP network interconnection exists between them) as if they were attached to a single LAN. As bridges are transparent, they do not appear in traceroute list, and no utility can make a distinction between a host working in one LAN and a host working in another LAN if these LANs are bridged (depending on the way the LANs are interconnected, latency and data rate between hosts may vary).

Network loops may emerge (intentionally or not) in complex topologies. Without any special treatment, loops would prevent network from functioning normally, as they would lead to avalanche-like packet multiplication. Each bridge runs an algorithm which calculates how the loop can be prevented. STP allows bridges to communicate with each other, so they can negotiate a loop free topology. All other alternative connections that would otherwise form loops, are put to standby, so that should the main connection fail, another connection could take its place. This algorithm exchange configuration messages (BPDU - Bridge Protocol Data Unit) periodically, so that all bridges would be updated with the newest information about changes in network topology. STP selects root bridge which is responosible for network reconfiguration, such as blocking and opening ports of the other bridges. The root bridge is the bridge with lowest bridge ID.

Additional Documents

http://ebtables.sourceforge.net/

Bridge Interface Setup

Home menu level: ***/interface bridge***

Description

To combine a number of networks into one bridge, a bridge interface should be created (later, all the desired interfaces should be set up as its ports). One MAC address will be assigned to all the bridged interfaces (the smallest MAC address will be chosen automatically).

Property Description

ageing-time (*time*; default: **5m**) - how long a host information will be kept in the bridge database

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol setting

forward-delay (*time*; default: **15s**) - time which is spent during the initialization phase of the bridge interface (i.e., after router startup or enabling the interface) in listening/learning state before the bridge will start functioning normally

garbage-collection-interval (*time*; default: **4s**) - how often to drop old (expired) host entries in the

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

bridge database. The garbage collection process expurges the entries older than defined by the ageing-time property

hello-time (*time*; default: **2s**) - how often send hello packets to other bridges

mac-address (*read-only: MAC address*) - MAC address for the interface

max-message-age (*time*; default: **20s**) - how long to remember Hello messages received from other bridges

mtu (*integer*; default: **1500**) - Maximum Transmission Unit

name (*name*; default: **bridgeN**) - a descriptive name of the bridge interface

priority (*integer*: 0..65535; default: **32768**) - bridge interface priority. The priority argument is used by Spanning Tree Protocol to determine, which port remains enabled if at least two ports form a loop

stp (*no* | *yes*; default: **no**) - whether to enable the Spanning Tree Protocol. Bridging loops will only be prevented if this property is turned on

Example

To add and enable a bridge interface that will forward all the protocols:

```
[admin@MikroTik] interface bridge> add; print
Flags: X - disabled, R - running
 0  R name="bridge1" mtu=1500 arp=enabled mac-address=61:64:64:72:65:73 stp=no
      priority=32768 ageing-time=5m forward-delay=15s
      garbage-collection-interval=4s hello-time=2s max-message-age=20s
[admin@MikroTik] interface bridge> enable 0
```

Port Settings

Home menu level: ***/interface bridge port***

Description

The submenu is used to enslave interfaces in a particular bridge interface.

Property Description

bridge (*name*; default: **none**) - the bridge interface the respective interface is grouped in

- **none** - the interface is not grouped in any bridge

interface (*read-only: name*) - interface name, which is to be included in a bridge

path-cost (*integer*: 0..65535; default: **10**) - path cost to the interface, used by STP to determine the 'best' path

priority (*integer*: 0..255; default: **128**) - interface priority compared to other interfaces, which are destined to the same network

Example

To group **ether1** and **ether2** in the already created **bridge1** bridge:

```
[admin@MikroTik] interface bridge port> set ether1,ether2 bridge=bridge1
[admin@MikroTik] interface bridge port> print
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
 # INTERFACE   BRIDGE PRIORITY PATH-COST
 0 ether1      bridge1   128      10
 1 ether2      bridge1   128      10
 2 wlan1       none      128      10
[admin@MikroTik] interface bridge port>
```

Bridge Monitoring

Command name: ***/interface bridge monitor***

Description

Used to monitor the current status of a bridge.

Property Description

bridge-id (*text*) - the bridge ID, which is in form of bridge-priority.bridge-MAC-address

designated-root (*text*) - ID of the root bridge

path-cost (*integer*) - the total cost of the path to the root-bridge

root-port (*name*) - port to which the root bridge is connected to

Example

To monitor a bridge:

```
[admin@MikroTik] interface bridge> monitor bridge1
          bridge-id: 32768.00:02:6F:01:CE:31
    designated-root: 32768.00:02:6F:01:CE:31
          root-port: ether2
          path-cost: 180

[admin@MikroTik] interface bridge>
```

Bridge Port Monitoring

Command name: ***/interface bridge port monitor***

Description

Statistics of an interface that belongs to a bridge

Property Description

designated-port (*text*) - port of designated-root bridge

designated-root (*text*) - ID of bridge, which is nearest to the root-bridge

port-id (*integer*) - port ID, which represents from port priority and port number, and is unique

status (*disabled* | *blocking* | *listening* | *learning* | *forwarding*) - the status of the bridge port:

- **disabled** - the interface is disabled. No frames are forwarded, no Bridge Protocol Data Units (BPDUs) are heard
- **blocking** - the port does not forward any frames, but listens for BPDUs

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **listening** - the port does not forward any frames, but listens to them
- **learning** - the port does not forward any frames, but learns the MAC addresses
- **forwarding** - the port forwards frames, and learns MAC addresses

Example

To monitor a bridge port:

```
[admin@MikroTik] interface bridge port> mo 0
               status: forwarding
              port-id: 28417
      designated-root: 32768.00:02:6F:01:CE:31
    designated-bridge: 32768.00:02:6F:01:CE:31
      designated-port: 28417
      designated-cost: 0
-- [Q quit|D dump|C-z pause]
```

Bridge Host Monitoring

Command name: ***/interface bridge host***

Property Description

age (*read-only: time*) - the time since the last packet was received from the host

bridge (*read-only: name*) - the bridge the entry belongs to

local (*read-only: flag*) - whether the host entry is of the bridge itself (that way all local interfaces are shown)

mac-address (*read-only: MAC address*) - host's MAC address

on-interface (*read-only: name*) - which of the bridged interfaces the host is connected to

Example

To get the active host table:

```
[admin@MikroTik] interface bridge host> print
Flags: L - local
   BRIDGE              MAC-ADDRESS       ON-INTERFACE       AGE
   bridge1             00:00:B4:5B:A6:58 ether1             4m48s
   bridge1             00:30:4F:18:58:17 ether1             4m50s
 L bridge1             00:50:08:00:00:F5 ether1             0s
 L bridge1             00:50:08:00:00:F6 ether2             0s
   bridge1             00:60:52:0B:B4:81 ether1             4m50s
   bridge1             00:C0:DF:07:5E:E6 ether1             4m46s
   bridge1             00:E0:C5:6E:23:25 prism1             4m48s
   bridge1             00:E0:F7:7F:0A:B8 ether1             1s
[admin@MikroTik] interface bridge host>
```

Bridge Firewall General Description

Home menu level: ***/interface bridge filter, /interface bridge nat, /interface bridge broute***

Description

The bridge firewall implements packet filtering and thereby provides security functions that are

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

used to manage data flow to, from and through bridge

Note that packets between bridged interfaces, just like any other IP traffic, are also passed through the 'generic' **/ip firewall** rules (but bridging filters are always applied before IP filters/NAT of the built-in chain of the same name, except for the **output** which is executed after IP Firewall Output). These rules can be used with real, physical receiving/transmitting interfaces, as well as with bridge interface that simply groups the bridged interfaces.

There are three bridge filter tables:

- **filter** - bridge firewall with three predefined chains:
  - **input** - filters packets, which destination is the bridge (including those packets that will be routed, as they are anyway destined to the bridge MAC address)
  - **output** - filters packets, which come from the bridge (including those packets that has been routed normally)
  - **forward** - filters packets, which are to be bridged (note: this chain is not applied to the packets that should be routed through the router, just to those that are traversing between the ports of the same bridge)
- **nat** - bridge network address translation provides ways for changing source/destination MAC addresses of the packets traversing a bridge. Has two built-in chains:
  - **scnat** - used for "hiding" a host or a network behind a different MAC address. This chain is applied to the packets leaving the router through a bridged interface
  - **dstnat** - used for redirecting some pakets to another destinations
- **broute** - makes bridge a brouter - router that performs routing on some of the packets, and bridging - on others. Has one predefined chain: **brouting**, which is traversed right after a packet enters an enslaved interface (before "Bridging Decision")

Note: the bridge destination NAT is executed before bridging desision

You can put packet marks in bridge firewall (filter, broute and NAT), which are the same as the packet marks in IP firewall put by mangle. So packet marks put by bridge firewall can be used in IP firewall, and vice versa

General bridge firewall properties are described in this section. Some parameters that differ between nat, broute and filter rules are described in further sections.

Property Description

802.3-sap (*integer*) - DSAP (Destination Service Access Point) and SSAP (Source Service Access Point) are 2 one byte fields, which identify the network protocol entities which use the link layer service. These bytes are always equal. Two hexadecimal digits may be specified here to match an SAP byte

802.3-type (*integer*) - Ethernet protocol type, placed after the IEEE 802.2 frame header. Works only if 802.3-sap is 0xAA (SNAP - Sub-Network Attachment Point header). For example, AppleTalk can be indicated by SAP code of 0xAA followed by a SNAP type code of 0x809B

arp-dst-address (*IP address*; default: **0.0.0.0/0**) - ARP destination address

arp-dst-mac-address (*MAC address*; default: **00:00:00:00:00:00**) - ARP destination MAC address

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

arp-hardware-type (*integer*; default: **1**) - ARP hardware type. This normally Ethernet (Type 1)

arp-opcode (*arp-nak* | *drarp-error* | *drarp-reply* | *drarp-request* | *inarp-request* | *reply* | *reply-reverse* | *request* | *request-reverse*) - ARP opcode (packet type)

- **arp-nak** - negative ARP reply (rarely used, mostly in ATM networks)
- **drarp-error** - Dynamic RARP error code, saying that an IP address for the given MAC address can not be allocated
- **drarp-reply** - Dynamic RARP reply, with a temporaty IP address assignment for a host
- **drarp-request** - Dynamic RARP request to assign a temporary IP address for the given MAC address
- **inarp-request** -
- **reply** - standard ARP reply with a MAC address
- **reply-reverse** - reverse ARP (RARP) reply with an IP address assigned
- **request** - standard ARP request to a known IP address to find out unknown MAC address
- **request-reverse** - reverse ARP (RARP) request to a known MAC address to find out unknown IP address (intended to be used by hosts to find out their own IP address, similarly to DHCP service)

arp-packet-type (*integer*) -

arp-src-address (*IP address*; default: **0.0.0.0/0**) - ARP source IP address

arp-src-mac-address (*MAC address*; default: **00:00:00:00:00:00**) - ARP source MAC address

chain (*text*) - bridge firewall chain, which the filter is functioning in (either a built-in one, or a user defined)

dst-address (*IP address*; default: **0.0.0.0/0**) - destination IP address (only if MAC protocol is set to IPv4)

dst-mac-address (*MAC address*; default: **00:00:00:00:00:00**) - destination MAC address

dst-port (*integer*: 0..65535) - destination port number or range (only for TCP or UDP protocols)

flow (*text*) - individual packet mark to match

in-bridge (*name*) - bridge interface through which the packet is coming in

in-interface (*name*) - physical interface (i.e., bridge port) through which the packet is coming in

ip-protocol (*ipsec-ah* | *ipsec-esp* | *ddp* | *egp* | *ggp* | *gre* | *hmp* | *idpr-cmtp* | *icmp* | *igmp* | *ipencap* | *encap* | *ipip* | *iso-tp4* | *ospf* | *pup* | *rspf* | *rdp* | *st* | *tcp* | *udp* | *vmtp* | *xns-idp* | *xtp*) - IP protocol (only if MAC protocol is set to IPv4)

- **ipsec-ah** - IPsec AH protocol
- **ipsec-esp** - IPsec ESP protocol
- **ddp** - datagram delivery protocol
- **egp** - exterior gateway protocol
- **ggp** - gateway-gateway protocol
- **gre** - general routing encapsulation
- **hmp** - host monitoring protocol
- **idpr-cmtp** - idpr control message transport
- **icmp** - internet control message protocol
- **igmp** - internet group management protocol

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **ipencap** - ip encapsulated in ip
- **encap** - ip encapsulation
- **ipip** - ip encapsulation
- **iso-tp4** - iso transport protocol class 4
- **ospf** - open shortest path first
- **pup** - parc universal packet protocol
- **rspf** - radio shortest path first
- **rdp** - reliable datagram protocol
- **st** - st datagram mode
- **tcp** - transmission control protocol
- **udp** - user datagram protocol
- **vmtp** - versatile message transport
- **xns-idp** - xerox ns idp
- **xtp** - xpress transfer protocol

jump-target (*name*) - if action=jump specified, then specifies the user-defined firewall chain to process the packet

limit (*integer* | *time* | *integer*) - restricts packet match rate to a given limit. Usefull to reduce the amount of log messages

- **Count** - maximum average packet rate, measured in packets per second (pps), unless followed by Time option
- **Time** - specifies the time interval over which the packet rate is measured
- **Burst** - number of packets to match in a burst

log-prefix (*text*) - defines the prefix to be printed before the logging information

mac-protocol (*integer* | *802.2* | *arp* | *ip* | *ipv6* | *ipx* | *rarp* | *vlan*) - Ethernet payload type (MAC-level protocol)

mark-flow (*name*) - marks existing flow

packet-type (*broadcast* | *host* | *multicast* | *other-host*) - MAC frame type:

- **broadcast** - broadcast MAC packet
- **host** - packet is destined to the bridge itself
- **multicast** - multicast MAC packet
- **other-host** - packet is destined to some other unicast address, not to the bridge itself

src-address (*IP address*; default: **0.0.0.0/0**) - source IP address (only if MAC protocol is set to IPv4)

src-mac-address (*MAC address*; default: **00:00:00:00:00:00**) - source MAC address

src-port (*integer*: 0..65535) - source port number or range (only for TCP or UDP protocols)

stp-flags (*topology-change* | *topology-change-ack*) - The BPDU (Bridge Protocol Data Unit) flags. Bridge exchange configuration messages named BPDU peridiocally for preventing from loop

- **topology-change** - topology change flag is set when a bridge detects port state change, to force all other bridges to drop their host tables and recalculate network topology
- **topology-change-ack** - topology change acknowledgement flag is sen in replies to the

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

notification packets

stp-forward-delay (*time*: 0..65535) - forward delay timer

stp-hello-time (*time*: 0..65535) - stp hello packets time

stp-max-age (*time*: 0..65535) - maximal STP message age

stp-msg-age (*time*: 0..65535) - STP message age

stp-port (*integer*: 0..65535) - stp port identifier

stp-root-address (*MAC address*) - root bridge MAC address

stp-root-cost (*integer*: 0..65535) - root bridge cost

stp-root-priority (*time*: 0..65535) - root bridge priority

stp-sender-address (*MAC address*) - stp message sender MAC address

stp-sender-priority (*integer*: 0..65535) - sender priority

stp-type (*config* | *tcn*) - the BPDU type

- **config** - configuration BPDU
- **tcn** - topology change notification

vlan-encap (*802.2* | *arp* | *ip* | *ipv6* | *ipx* | *rarp* | *vlan*) - the MAC protocol type encapsulated in the VLAN frame

vlan-id (*integer*: 0..4095) - VLAN identifier field

vlan-priority (*integer*: 0..7) - the user priority field

Notes

stpmatchers are only valid if destination MAC address is 01:80:C2:00:00:00/FF:FF:FF:FF:FF:FF (Bridge Group address), also **stp** should be enabled.

ARP matchers are only valid if **mac-protocol** is **arp** or **rarp**

VLAN matchers are only valid for **vlan** ethernet protocol

IP-related matchers are only valid if **mac-protocol** is set as **ipv4**

802.3 matchers are only consulted if the actual frame is compliant with IEEE 802.2 and IEEE 802.3 standards (**note**: it is not the industry-standard Ethernet frame format used in most networks worldwide!). These matchers are ignored for other packets.

Bridge Packet Filter

Home menu level: ***/interface bridge filter***

Description

This section describes bridge packet filter specific filtering options, which were omitted in the general firewall description

Property Description

action (*accept* | *drop* | *jump* | *log* | *mark* | *passthrough* | *return*; default: **accept**) - action to undertake

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

if the packet matches the rule, one of the:

- **accept** - accept the packet. No action, i.e., the packet is passed through without undertaking any action, and no more rules are processed in the relevant list/chain
- **drop** - silently drop the packet (without sending the ICMP reject message)
- **jump** - jump to the chain specified by the value of the jump-target argument
- **log** - log the packet
- **mark** - mark the packet to use the mark later
- **passthrough** - ignore this rule and go on to the next one. Acts the same way as a disabled rule, except for ability to count packets
- **return** - return to the previous chain, from where the jump took place

out-bridge (*name*) - outgoing bridge interface

out-interface (*name*) - interface via packet is leaving the bridge

Bridge NAT

Home menu level: ***/interface bridge nat***

Description

This section describes bridge NAT options, which were omitted in the general firewall description

Property Description

action (*accept* | *arp-reply* | *drop* | *dst-nat* | *jump* | *log* | *mark* | *passthrough* | *redirect* | *return* | *src-nat*; default: **accept**) - action to undertake if the packet matches the rule, one of the:

- **accept** - accept the packet. No action, i.e., the packet is passed through without undertaking any action, and no more rules are processed in the relevant list/chain
- **arp-reply** - send a reply to an ARP request (any other packets will be ignored by this rule) with the specified MAC address (only valid in dstnat chain)
- **drop** - silently drop the packet (without sending the ICMP reject message)
- **dst-nat** - change destination MAC address of a packet (only valid in dstnat chain)
- **jump** - jump to the chain specified by the value of the jump-target argument
- **log** - log the packet
- **mark** - mark the packet to use the mark later
- **passthrough** - ignore this rule and go on to the next one. Acts the same way as a disabled rule, except for ability to count packets
- **redirect** - redirect the packet to the bridge itself (only valid in dstnat chain)
- **return** - return to the previous chain, from where the jump took place
- **src-nat** - change source MAC address of a packet (only valid in srcnat chain)

out-bridge (*name*) - outgoing bridge interface

out-interface (*name*) - interface via packet is leaving the bridge

to-arp-reply-mac-address (*MAC address*) - source MAC address to put in Ethernet frame and ARP payload, when action=arp-reply is selected

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

to-dst-mac-address (*MAC address*) - destination MAC address to put in Ethernet frames, when action=dst-nat is selected

to-src-mac-address (*MAC address*) - source MAC address to put in Ethernet frames, when action=src-nat is selected

Bridge Brouting Facility

Home menu level: ***/interface bridge broute***

Description

This section describes broute facility specific options, which were omitted in the general firewall description

The Brouting table is applied to every packet entering a forwarding enslaved interface (i.e., it does not work on regular interfaces, which are not included in a bridge)

Property Description

action (*accept* | *drop* | *dst-nat* | *jump* | *log* | *mark* | *passthrough* | *redirect* | *return*; default: **accept**) - action to undertake if the packet matches the rule, one of the:

- **accept** - let the bridging code decide, what to do with this packet
- **drop** - extract the packet from bridging code, making it appear just like it would come from a not-bridged interface (no further bridge decisions or filters will be applied to this packet except if the packet would be router out to a bridged interface, in which case the packet would be processed normally, just like any other routed packet)
- **dst-nat** - change destination MAC address of a packet (only valid in dstnat chain), an let bridging code to decide further actions
- **jump** - jump to the chain specified by the value of the jump-target argument
- **log** - log the packet
- **mark** - mark the packet to use the mark later
- **passthrough** - ignore this rule and go on to the next one. Acts the same way as a disabled rule, except for ability to count packets
- **redirect** - redirect the packet to the bridge itself (only valid in dstnat chain), an let bridging code to decide further actions
- **return** - return to the previous chain, from where the jump took place

to-dst-mac-address (*MAC address*) - destination MAC address to put in Ethernet frames, when action=dst-nat is selected

Troubleshooting

Description

- **Router shows that my rule is invalid**
  - in-interface, in-bridge (or in-bridge-port) is specified, but such an interface does not exist

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- there is an action=mark-packet, but no new-packet-mark
- there is an action=mark-connection, but no new-connection-mark
- there is an action=mark-routing, but no new-routing-mark

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

CISCO/Aironet 2.4GHz 11Mbps Wireless Interface

Document revision 1.2 (Mon May 31 20:18:58 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Additional Documents
Wireless Interface Configuration
  Description
  Property Description
  Example
  Example
Troubleshooting
  Description
Application Examples
  Point-to-Multipoint Wireless LAN
  Point-to-Point Wireless LAN

General Information

Summary

The MikroTik RouterOS supports the following CISCO/Aironet 2.4GHz Wireless ISA/PCI/PC Adapter hardware:

- Aironet ISA/PCI/PC4800 2.4GHz DS 11Mbps Wireless LAN Adapters (100mW)
- Aironet ISA/PCI/PC4500 2.4GHz DS 2Mbps Wireless LAN Adapters (100mW)
- CISCO AIR-PCI340 2.4GHz DS 11Mbps Wireless LAN Adapters (30mW)
- CISCO AIR-PCI/PC350/352 2.4GHz DS 11Mbps Wireless LAN Adapters (100mW)

Specifications

Packages required: ***wireless***
License required: ***level4***
Home menu level: ***/interface pc***
Standards and Technologies: ***IEEE802.11b***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***Device Driver List***
- ***IP Addresses and ARP***
- ***Log Management***

Additional Documents

- ***CISCO Aironet 350 Series***

For more information about the CISCO/Aironet PCI/ISA adapter hardware please see the relevant User's Guides and Technical Reference Manuals in PDF format:

- ***710-003638a0.pdf***for PCI/ISA 4800 and 4500 series adapters
- ***710-004239B0.pdf***for PC 4800 and 4500 series adapters

Documentation about CISCO/Aironet Wireless Bridges and Access Points can be found in archives:

- ***AP48MAN.exe*** for AP4800 Wireless Access Point
- ***BR50MAN.exe*** for BR500 Wireless Bridge

Wireless Interface Configuration

Home menu level: ***/interface pc***

Description

CISCO/Aironet 2.4GHz card is an interface for wireless networks operating in IEEE 802.11b standard. If the wireless interface card is not registered to an AP, the green status led is blinking fast. If the wireless interface card is registered to an AP, the green status led is blinking slow. To set the wireless interface for working with an access point (register to the AP), typically you should set the following parameters:

- The **service set identifier**. It should match the ssid of the AP. Can be blank, if you want the wireless interface card to register to an AP with any ssid. The ssid will be received from the AP, if the AP is broadcasting its ssid.
- The data-rate of the card should match one of the supported data rates of the AP. Data rate 'auto' should work in most cases.

Loading the Driver for the Wireless Adapter

PCI and PC (PCMCIA) cards do not require a 'manual' driver loading, since they are recognized automatically by the system and the driver is loaded at the system startup.

The ISA card requires the driver to be loaded by issuing the following command:

There can be several reasons for a failure to load the driver:

- **The driver cannot be loaded because other device uses the requested IRQ.**
  Try to set different IRQ using the DIP switches.
- **The requested I/O base address cannot be used on your motherboard**

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Try to change the I/O base address using the DIP switches

Property Description

ap1 (*MAC address*) - forces association to the specified access point

ap2 (*MAC address*) - forces association to the specified access point

ap3 (*MAC address*) - forces association to the specified access point

ap4 (*MAC address*) - forces association to the specified access point

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol

beacon-period (*integer*: 20..976; default: **100**) - Specifies beaconing period (applicable to ad-hoc mode only)

card-type (*read-only: text*) - your CISCO/Aironet adapter model and type

client-name (*text*; default: **""**) - client name

data-rate (*1Mbit/s* | *2Mbit/s* | *5.5Mbit/s* | *11Mbit/s* | *auto*; default: **1Mbit/s**) - data rate in Mbit/s

fragmentation-threshold (*integer*: 256..2312; default: **2312**) - this threshold controls the packet size at which outgoing packets will be split into multiple fragments. If a single fragment transmit error occurs, only that fragment will have to be retransmitted instead of the whole packet. Use a low setting in areas with poor communication or with a great deal of radio interference

frequency - Channel Frequency in MHz (applicable to ad-hoc mode only)

join-net (*time*; default: **10**) - an amount of time,during which the interface operating in ad-hoc mode will try to connect to an existing network rather than create a new one

- **0** - do not create own network

long-retry-limit (*integer*: 0..128; default: **16**) - specifies the number of times an unfragmented packet is retried before it is dropped

mode (*infrastructure* | *ad-hoc*; default: **infrastructure**) - operation mode of the card

modulation (*cck* | *default* | *mbok*; default: **cck**) - modulation mode

- **cck** - Complementary Code Keying
- **mbok** - M-ary Bi-Orthogonal Keying

mtu (*integer*: 256..2048; default: **1500**) - Maximum Transmission Unit

name (*name*) - descriptive interface name

rts-threshold (*integer*: 0..2312; default: **2312**) - determines the packet size at which the interface issues a request to send (RTS) before sending the packet. A low value can be useful in areas where many clients are associating with the access point or bridge, or in areas where the clients are far apart and can detect only the access point or bridge and not each other

rx-antenna (*both* | *default* | *left* | *right*; default: **both**) - receive antennas

short-retry-limit (*integer*: 0..128; default: **16**) - specifies the number of times a fragmented packet is retried before it is dropped

ssid1 (*text*; default: **tsunami**) - establishes the adapter's service set identifier This value must match the SSID of the system in order to operate in infrastructure mode

ssid2 (*text*; default: **""**) - service set identifier 2

ssid3 (*text*; default: **""**) - service set identifier 3

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

tx-antenna (*both* | *default* | *left* | *right*; default: **both**) - transmit antennas

tx-power (*1* | *5* | *20* | *50* | *100*; default: **100**) - transmit power in mW

world-mode (yes | no; default: **no**) - if set, client adapter automatically inherit channel configuration properties directly from the access point to which they associate. This feature enables a user to use a client adapter around the world while still maintaining regulatory compliance

Example

Interface informational printouts

```
[admin@MikroTik] > interface print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                TYPE            MTU
  0  R ether1               ether           1500
  1  X ether2               ether           1500
  2  X pc1                  pc              1500
[admin@MikroTik] interface> set 2 name aironet
[admin@MikroTik] interface> enable aironet
[admin@MikroTik] > interface print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                TYPE            MTU
  0  R ether1               ether           1500
  1  X ether2               ether           1500
  2  R aironet              pc              1500
[admin@MikroTik] > interface pc
[admin@MikroTik] interface pc> print
Flags: X - disabled, R - running
  0  R name="aironet" mtu=1500 mac-address=00:40:96:29:2F:80 arp=enabled
       client-name="" ssid1="tsunami" ssid2="" ssid3="" mode=infrastructure
       data-rate=1Mbit/s frequency=2437MHz modulation=cck tx-power=100
       ap1=00:00:00:00:00:00 ap2=00:00:00:00:00:00 ap3=00:00:00:00:00:00
       ap4=00:00:00:00:00:00 rx-antenna=right tx-antenna=right beacon-period=100
       long-retry-limit=16 short-retry-limit=16 rts-threshold=2312
       fragmentation-threshold=2312 join-net=10s card-type=PC4800A 3.65

[admin@MikroTik] interface pc>
```

Interface status monitoring

```
[admin@MikroTik] interface pc> monitor 0
         synchronized: no
           associated: no
         error-number: 0

[admin@MikroTik] interface pc>
```

Example

Suppose we want to configure the wireless interface to accomplish registration on the AP with a **ssid** 'mt'.

We need to change the value of ssid property to the corresponding value.

To view the results, we can use **monitor** feature.

```
[admin@MikroTik] interface pc> set 0 ssid1 mt
[admin@MikroTik] interface pc> monitor 0
         synchronized: yes
           associated: yes
            frequency: 2412MHz
            data-rate: 11Mbit/s
                 ssid: "mt"
        access-point: 00:02:6F:01:5D:FE
    access-point-name: ""
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
        signal-quality: 132
       signal-strength: -82
          error-number: 0
[admin@MikroTik] interface pc>
```

Troubleshooting

Description

Keep in mind, that not all combinations of I/O base addresses and IRQs may work on particular motherboard. It is recommended that you choose an IRQ not used in your system, and then try to find an acceptable I/O base address setting. As it has been observed, the IRQ 5 and I/O 0x300 or 0x180 will work in most cases.

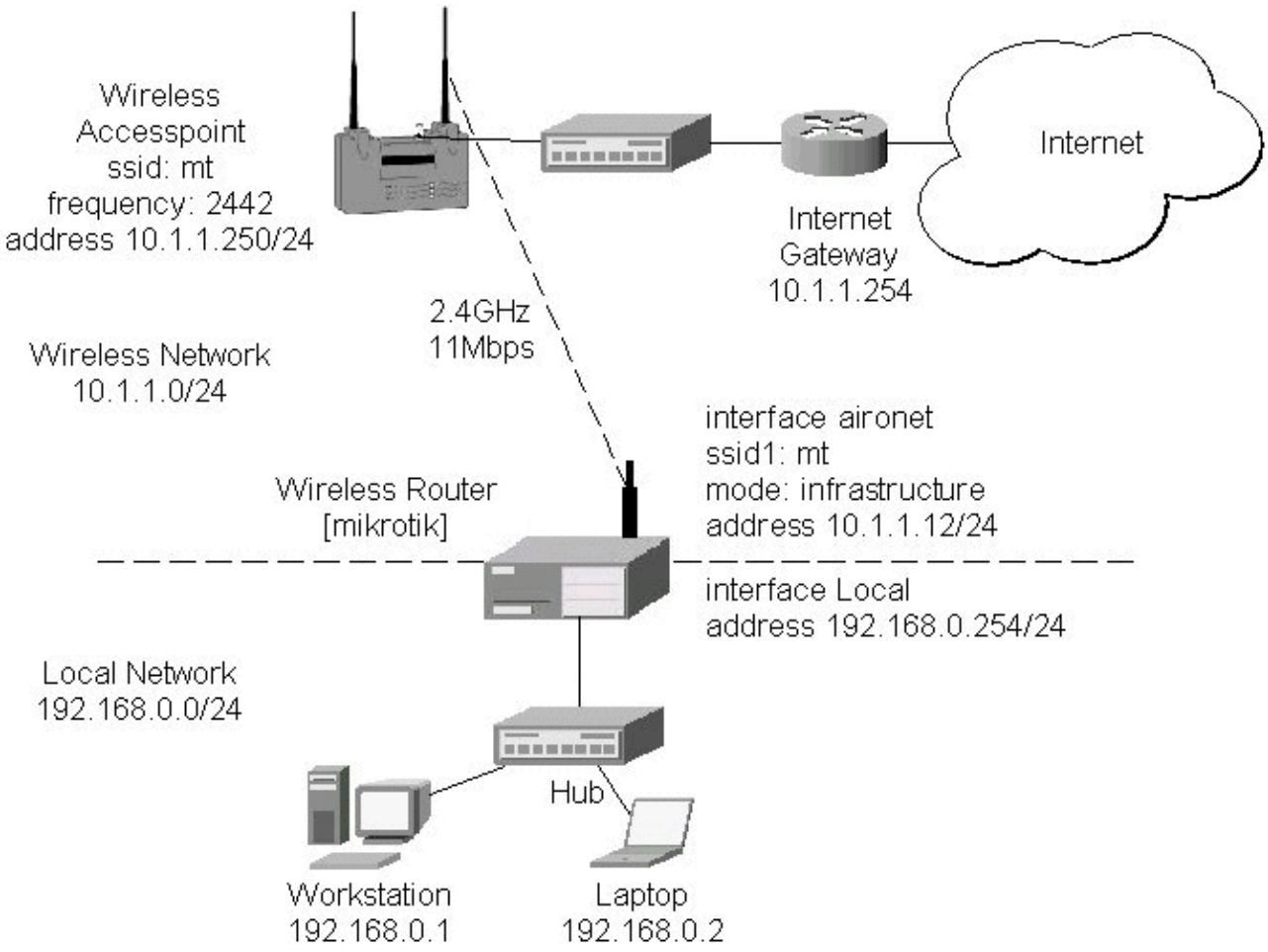
- **The driver cannot be loaded because other device uses the requested IRQ.**
  Try to set different IRQ using the DIP switches.
- **The requested I/O base address cannot be used on your motherboard.**
  Try to change the I/O base address using the DIP switches.
- **The pc interface does not show up under the interfaces list**
  Obtain the required license for 2.4/5GHz Wireless Client feature.
- **The wireless card does not register to the Access Point**
  Check the cabling and antenna alignment.

Application Examples

Point-to-Multipoint Wireless LAN

Let us consider the following network setup with CISCO/Aironet Wireless Access Point as a base station and MikroTik Wireless Router as a client:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



The access point is connected to the wired network's HUB and has IP address from the network 10.1.1.0/24.

The minimum configuration required for the AP is:

1. Setting the Service Set Identifier (up to 32 alphanumeric characters). In our case we use ssid "mt".
2. Setting the allowed data rates at 1-11Mbps, and the basic rate at 1Mbps.
3. Choosing the frequency, in our case we use 2442MHz.
4. (For CISCO/Aironet Bridges only) Set Configuration/Radio/Extended/Bridge/mode=access_point. If you leave it to 'bridge_only', it wont register clients.
5. Setting the identity parameters Configuration/Ident: Inaddr, Inmask, and Gateway. These are required if you want to access the AP remotely using telnet or http.

The IP addresses assigned to the wireless interface should be from the network 10.1.1.0/24:

```
[admin@MikroTik] ip address> add address 10.1.1.12/24 interface aironet
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.1.1.12/24       10.1.1.0        10.1.1.255      aironet
  1   192.168.0.254/24   192.168.0.0     192.168.0.255   Local
[admin@MikroTik] ip address>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

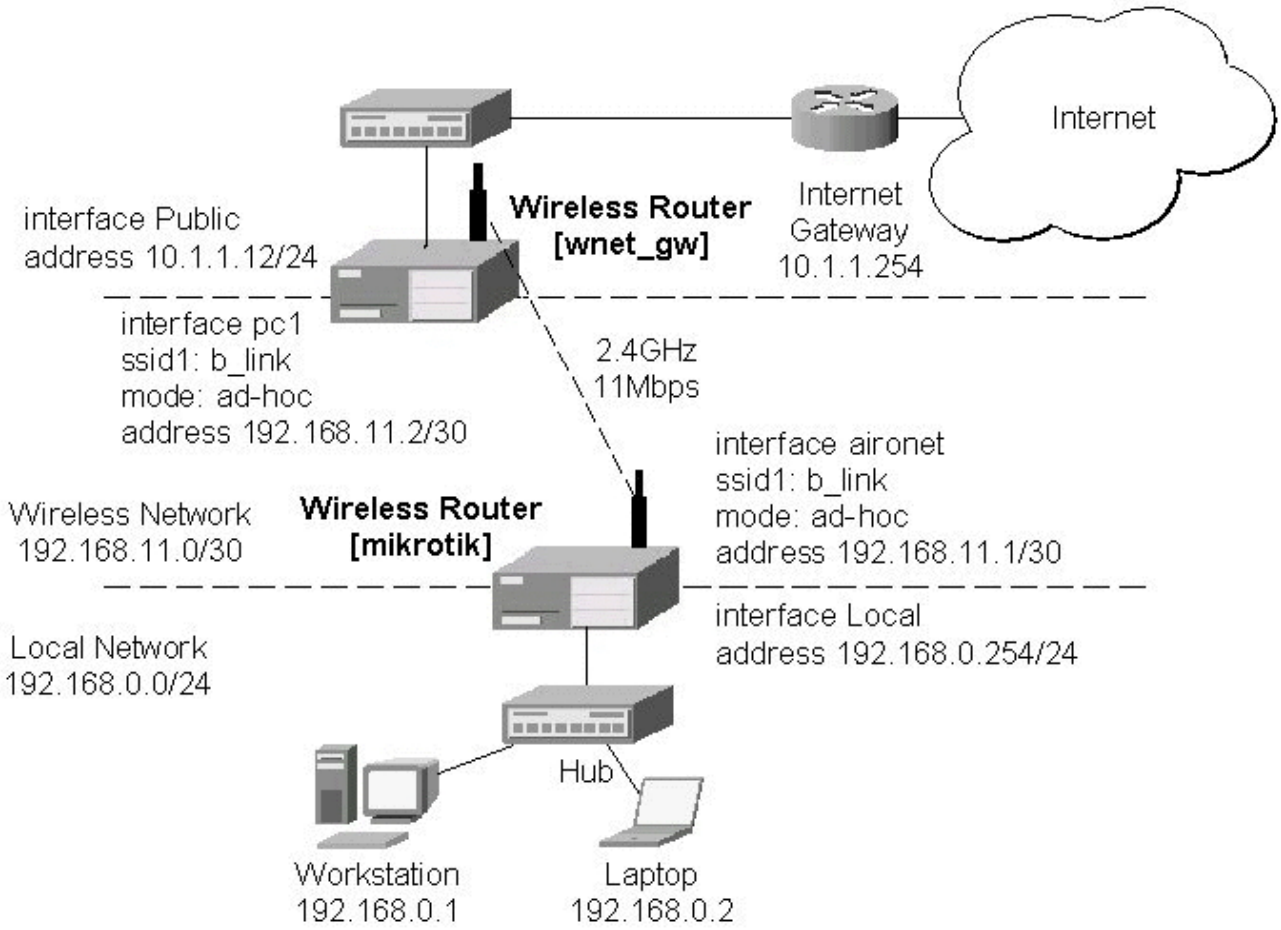
The default route should be set to the gateway router 10.1.1.254 (! not the AP 10.1.1.250 !):

```
[admin@MikroTik] ip route> add gateway=10.1.1.254
[admin@MikroTik] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0  S 0.0.0.0/0          r 10.1.1.254      1        aironet
    1 DC 192.168.0.0/24     r 0.0.0.0         0        Local
    2 DC 10.1.1.0/24        r 0.0.0.0         0        aironet

[admin@MikroTik] ip route>
```

Point-to-Point Wireless LAN

Point-to-Point links provide a convenient way to connect a pair of clients on a short distance.

Let us consider the following point-to-point wireless network setup with two MikroTik wireless routers:



To establish a point-to-point link, the configuration of the wireless interface should be as follows:

- A unique Service Set Identifier should be chosen for both ends, say "mt"
- A channel frequency should be selected for the link, say 2412MHz
- The operation mode should be set to ad-hoc

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- One of the units (slave) should have wireless interface property join-net set to 0s (never create a network), the other unit (master) should be set to 1s or whatever, say 10s. This will enable the master unit to create a network and register the slave unit to it.

The following command should be issued to change the settings for the pc interface of the master unit:

```
[admin@MikroTik] interface pc> set 0 mode=ad-hoc ssid1=mt frequency=2442MHz \
\... bitrate=auto
[admin@MikroTik] interface pc>
```

For 10 seconds (this is set by the property **join-net**) the wireless card will look for a network to join. The status of the card is not synchronized, and the green status light is blinking fast. If the card cannot find a network, it creates its own network. The status of the card becomes synchronized, and the green status led becomes solid.

The monitor command shows the new status and the MAC address generated:

```
[admin@MikroTik] interface pc> monitor 0
         synchronized: yes
           associated: yes
            frequency: 2442MHz
            data-rate: 11Mbit/s
                 ssid: "mt"
         access-point: 2E:00:B8:01:98:01
    access-point-name: ""
       signal-quality: 35
      signal-strength: -62
         error-number: 0
[admin@MikroTik] interface pc>
```

The other router of the point-to-point link requires the operation mode set to **ad-hoc**, the System Service Identifier set to 'mt', and the channel frequency set to 2412MHz. If the cards are able to establish RF connection, the status of the card should become synchronized, and the green status led should become solid immediately after entering the command:

```
[admin@wnet_gw] interface pc> set 0 mode=ad-hoc ssid1=b_link frequency=2412MHz \
\... bitrate=auto
[admin@wnet_gw] interface pc> monitor 0
         synchronized: yes
           associated: no
            frequency: 2442MHz
            data-rate: 11Mbit/s
                 ssid: "b_link"
         access-point: 2E:00:B8:01:98:01
    access-point-name: ""
       signal-quality: 131
      signal-strength: -83
         error-number: 0

[admin@wnet_gw] interface pc>
```

As we see, the MAC address under the **access-point** property is the same as on the first router.

If desired, IP addresses can be assigned to the wireless interfaces of the pint-to-point linked routers using a smaller subnet, say 30-bit one:

```
[admin@MikroTik] ip address> add address 192.168.11.1/30 interface aironet
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   192.168.11.1/30    192.168.11.0    192.168.11.3    aironet
  1   192.168.0.254/24   192.168.0.0     192.168.0.255   Local
[admin@MikroTik] ip address>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

The second router will have address 192.168.11.2. The network connectivity can be tested by using ping or bandwidth test:

```
[admin@wnet_gw] ip address> add address 192.168.11.2/30 interface aironet
[admin@wnet_gw] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   192.168.11.2/30    192.168.11.0    192.168.11.3    aironet
  1   10.1.1.12/24       10.1.1.0        10.1.1.255      Public
[admin@wnet_gw] ip address> /ping 192.168.11.1
192.168.11.1 pong: ttl=255 time=3 ms
192.168.11.1 pong: ttl=255 time=1 ms
192.168.11.1 pong: ttl=255 time=1 ms
192.168.11.1 pong: ttl=255 ping interrupted
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 1/1.5/3 ms
[admin@wnet_gw] interface pc> /tool bandwidth-test 192.168.11.1 protocol tcp
                  status: running
              rx-current: 4.61Mbps
    rx-10-second-average: 4.25Mbps
        rx-total-average: 4.27Mbps

[admin@wnet_gw] interface pc> /tool bandwidth-test 192.168.11.1 protocol udp size 1500
                  status: running
              rx-current: 5.64Mbps
    rx-10-second-average: 5.32Mbps
        rx-total-average: 4.87Mbps

[admin@wnet_gw] interface pc>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Cyclades PC300 PCI Adapters

Document revision 1.1 (Fri Mar 05 08:13:30 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
Synchronous Interface Configuration
  Description
  Property Description
Troubleshooting
  Description
RSV/V.35 Synchronous Link Applications
  Example

General Information

Summary

The MikroTik RouterOS supports the following Cyclades PC300 Adapter hardware:

- RSV/V.35 (RSV models) with 1 or 2 RS-232/V.35 interfaces on standard DB25/M.34 connector, 5Mbps, internal or external clock
- T1/E1 (TE models) with 1 or 2 T1/E1/G.703 interfaces on standard RJ48C connector, Full/Fractional, internal or external clock
- X.21 (X21 models) with 1 or 2 X.21 on standard DB-15 connector, 8Mbps, internal or external clock

Specifications

Packages required: ***synchronous***
License required: ***level4***
Home menu level: ***/interface cyclades***
Standards and Technologies: ***X.21, X.35, T1/E1/G.703, Frame Relay, PPP, Cisco-HDLC***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***Log Management***

Synchronous Interface Configuration

Home menu level: ***/interface cyclades***

Description

You can install up to four Cyclades PC300 PCI Adapters in one PC box, if you have so many adapter slots and IRQs available.

The Cyclades PC300/RSV Synchronous PCI Adapter comes with a V.35 cable. This cable should work for all standard modems, which have V.35 connections. For synchronous modems, which have a DB-25 connection, you should use a standard DB-25 cable.

Connect a communication device, e.g., a baseband modem, to the V.35 port and turn it on. The MikroTik driver for the Cyclades Synchronous PCI Adapter allows you to unplug the V.35 cable from one modem and plug it into another modem with a different clock speed, and you do not need to restart the interface or router.

Property Description

name (*name*; default: **cycladesN**) - descriptive interface name

mtu (*integer*; default: **1500**) - Maximum Transmission Unit for the interface

line-protocol (*cisco-hdlc* | *frame-relay* | *sync-ppp*; default: **sync-ppp**) - line protocol

media-type (*E1* | *T1* | *V24* | *V35* | *X21*; default: **V35**) - the hardware media used for this interface

clock-rate (*integer*; default: **64000**) - internal clock rate in bps

clock-source (*internal* | *external* | *tx-internal*; default: **external**) - source clock

line-code (*AMI* | *B8ZS* | *HDB3* | *NRZ*; default: **B8ZS**) - for T1/E1 channels only. Line modulation method:

- **AMI** - Alternate Mark Inversion
- **B8ZS** - Binary 8-Zero Substitution
- **HDB3** - High Density Bipolar 3 Code (ITU-T)
- **NRZ** - Non-Return-To-Zero

framing mode (*CRC4* | *D4* | *ESF* | *Non-CRC4* | *Unframed*; default: **ESF**) - for T1/E1 channels only. The frame mode:

- **CRC4** - Cyclic Redundancy Check 4-bit (E1 Signaling, Europe)
- **D4** - Fourth Generation Channel Bank (48 Voice Channels on 2 T-1s or 1 T-1c)
- **ESF** - Extended Superframe Format
- **Non-CRC4** - plain Cyclic Redundancy Check
- **Unframed** - do not check frame integrity

line-build-out (*0dB* | *7.5dB* | *15dB* | *22.5dB*; default: **0**) - for T1 channels only. Line Build Out Signal Level.

rx-sensitivity (*long-haul* | *short-haul*; default: **short-haul**) - for T1/E1 channels only. Numbers of active channels (up to 32 for E1 and up to 24 for T1)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

chdlc-keepalive (*time*; default: **10s**) - Cisco-HDLC keepalive interval in seconds

frame-relay-dce (*yes* | *no*; default: **no**) - specifies whether the device operates in Data Communication Equipment mode. The value yes is suitable only for T1 models

frame-relay-lmi-type (*ansi* | *ccitt*; default: **ansi**) - Frame Relay Line Management Interface Protocol type

Troubleshooting

Description

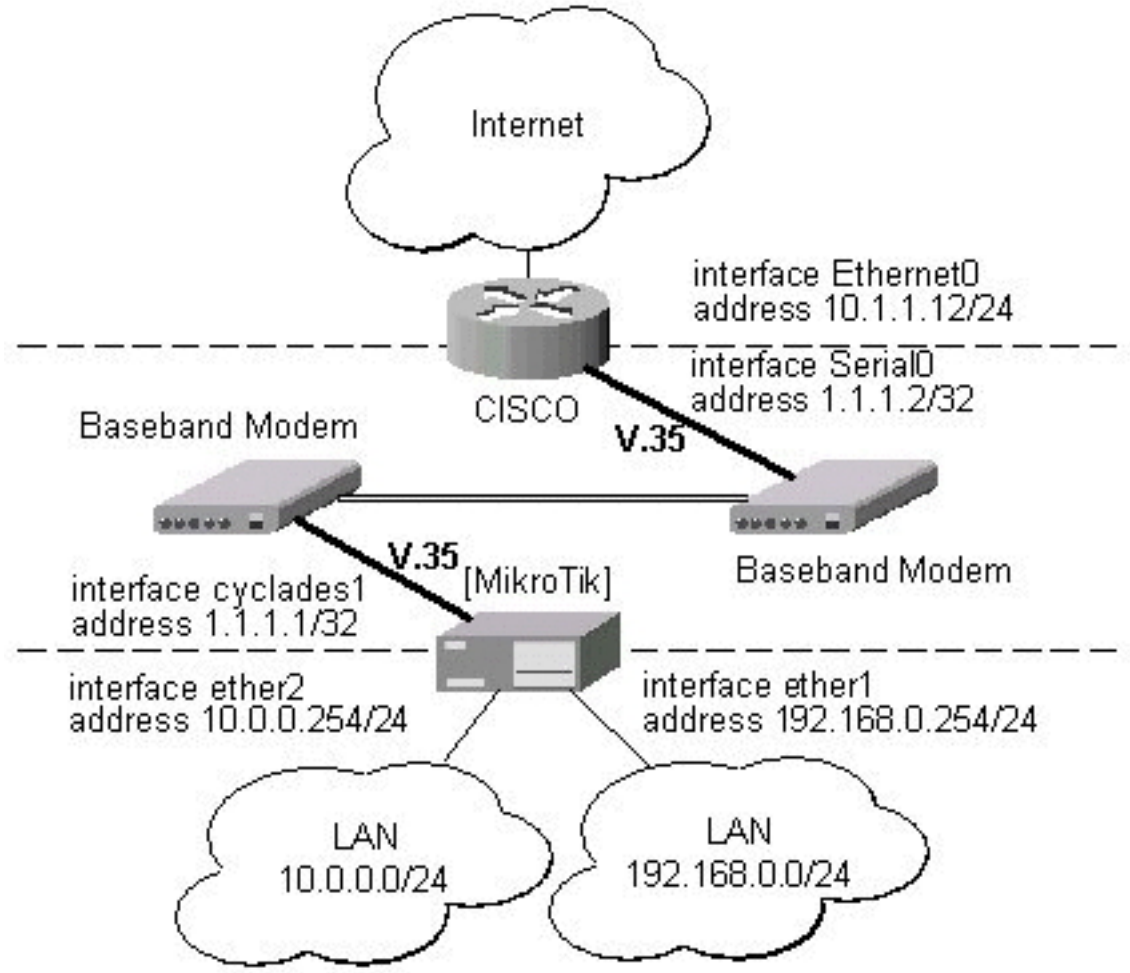
- **The cyclades interface does not show up under the interfaces list**
  Obtain the required license for synchronous feature
- **The synchronous link does not work**
  Check the V.35 cabling and the line between the modems. Read the modem manual

RSV/V.35 Synchronous Link Applications

Example

Let us consider the following network setup with MikroTik Router connected to a leased line with baseband modems and a CISCO router at the other end:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



The driver for the Cyclades PC300/RSV Synchronous PCI Adapter should load automatically. The interface should be enabled according to the instructions given above. The **IP addresses** assigned to the cyclades interface should be as follows:

```
[admin@MikroTik] ip address> add address=1.1.1.1/32 interface=cyclades1
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.219/24      10.0.0.0        10.0.0.255      ether1
  1   1.1.1.1/32         1.1.1.1         1.1.1.1         cyclades1
  2   192.168.0.254/24   192.168.0.0     192.168.0.255   ether2
[admin@MikroTik] ip address> /ping 1.1.1.2
1.1.1.2 64 byte pong: ttl=255 time=12 ms
1.1.1.2 64 byte pong: ttl=255 time=8 ms
1.1.1.2 64 byte pong: ttl=255 time=7 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 7/9.0/12 ms
[admin@MikroTik] ip address> /tool flood-ping 1.1.1.2 size=1500 count=50
        sent: 50
    received: 50
     min-rtt: 1
     avg-rtt: 1
     max-rtt: 9

[admin@MikroTik] ip address>
```

Note that for the point-to-point link the network mask is set to 32 bits, the argument **network** is set to the **IP address** of the other end, and the broadcast address is set to 255.255.255.255. The default route should be set to gateway router 1.1.1.2:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] ip route> add gateway 1.1.1.2 interface cyclades1
[admin@MikroTik] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0  S 0.0.0.0/0          r 1.1.1.2         1        cyclades1
    1 DC 10.0.0.0/24        r 0.0.0.0         0        ether1
    2 DC 192.168.0.0/24     r 0.0.0.0         0        ether2
    3 DC 1.1.1.2/32         r 0.0.0.0         0        cyclades1
[admin@MikroTik] ip route>
```

The configuration of the CISCO router at the other end (part of the configuration) is:

```
CISCO#show running-config
Building configuration...

Current configuration:
...
!
interface Ethernet0
 description connected to EthernetLAN
 ip address 10.1.1.12 255.255.255.0
!
interface Serial0
 description connected to MikroTik
 ip address 1.1.1.2 255.255.255.252
 serial restart-delay 1
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.1.1.254
!
...
end

CISCO#

Send ping packets to the MikroTik router:


CISCO#ping 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/32/40 ms
CISCO#
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Ethernet Interfaces

Document revision 1.2 (Fri Apr 16 12:35:37 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Additional Documents
Ethernet Interface Configuration
  Property Description
  Notes
  Example
Monitoring the Interface Status
  Property Description
  Notes
  Example
Troubleshooting
  Description

General Information

Summary

MikroTik RouterOS supports various types of Ethernet Interfaces. The complete list of supported Ethernet NICs can be found in the ***Device Driver List***.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/interface ethernet***
Standards and Technologies: ***IEEE 802.3***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***
- ***DHCP Client and Server***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Additional Documents

- ***http://www.ethermanage.com/ethernet/ethernet.html***
- ***http://www.dcs.gla.ac.uk/~liddellj/nct/ethernet_protocol.html***

Ethernet Interface Configuration

Home menu level: ***/interface ethernet***

Property Description

name (*name*; default: **etherN**) - assigned interface name, whrere 'N' is the number of the ethernet interface

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol

cable-setting (*default* | *short* | *standard*; default: **default**) - changes the cable length setting (only applicable to NS DP83815/6 cards)

- **default** - suport long cables
- **short** - support short cables
- **standard** - same as default

mtu (*integer*; default: **1500**) - Maximum Transmission Unit

disable-running-check (*yes* | *no*; default: **yes**) - disable running check. If this value is set to 'no', the router automatically detects whether the NIC is connected with a device in the network or not

mac-address (*MAC address*) - set the Media Access Control number of the card

auto-negotiation (*yes* | *no*; default: **yes**) - when enabled, the interface "advertises" its maximum capabilities to achieve the best connection possible

full-duplex (*yes* | *no*; default: **yes**) - defines whether the transmission of data appears in two directions simultaneously

speed (*10 Mbps* | *100 Mbps* | *1 Gbps*) - sets the data transmission speed of the interface. By default, this value is the maximal data rate supported by the interface

Notes

For some Ethernet NICs it is possible to blink the LEDs for 10s. Type **/interface ethernet blink ether1** and watch the NICs to see the one which has blinking LEDs.

When **disable-running-check** is set to **no**, the router automatically detects whether the NIC is connected to a device in the network or not. When the remote device is not connected (the leds are not blinking), the route which is set on the specific interface, becomes invalid.

Example

```
[admin@MikroTik] > interface print
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE    MTU
 0  X ether1                      ether           0          0          1500
[admin@MikroTik] > interface enable ether1
[admin@MikroTik] > interface print
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
Flags: X - disabled, D - dynamic, R - running
 #    NAME                        TYPE            RX-RATE    TX-RATE    MTU
 0  R ether1                      ether           0          0          1500
[admin@MikroTik] > interface ethernet
[admin@MikroTik] interface ethernet> print
Flags: X - disabled, R - running
 #    NAME                                  MTU   MAC-ADDRESS       ARP
 0  R ether1                                1500  00:0C:42:03:00:F2 enabled
[admin@MikroTik] interface ethernet> print detail
Flags: X - disabled, R - running
 0  R name="ether1" mtu=1500 mac-address=00:0C:42:03:00:F2 arp=enabled
      disable-running-check=yes auto-negotiation=yes full-duplex=yes
      cable-settings=default speed=100Mbps
[admin@MikroTik] interface ethernet>
```

Monitoring the Interface Status

Command name: ***/interface ethernet monitor***

Property Description

status (*link-ok* | *no-link* | *unknown*) - status of the interface, one of the:

- **link-ok** - the card has connected to the network
- **no-link** - the card has not connected to the network
- **unknown** - the connection is not recognized

rate (*10 Mbps* | *100 Mbps* | *1 Gbps*) - the actual data rate of the connection

auto-negotiation (*done* | *incomplete*) - fast link pulses (FLP) to the adjacent link station to negotiate the SPEED and MODE of the link

- **done** - negotiation done
- **incomplete** - negotiation failed

full-duplex (*yes* | *no*) - whether transmission of data occurs in two directions simultaneously

Notes

See the ***IP Addresses and ARP*** section of the manual for information how to add **IP addresses** to the interfaces.

Example

```
[admin@MikroTik] interface ethernet> monitor ether1,ether2
             status: link-ok link-ok
    auto-negotiation: done    done
                rate: 100Mbps 100Mbps
         full-duplex: yes     yes
```

Troubleshooting

Description

- **Interface monitor shows wrong information**
  In some very rare cases it is possible that the device driver does not show correct information,

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

but it does not affect the NIC's performance (of course, if your card is not broken)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

FarSync X.21 Interface

Document revision 1.1 (Fri Mar 05 08:14:24 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Additional Documents
Synchronous Interface Configuration
  Description
  Property Description
  Example
Troubleshooting
  Description
Synchronous Link Applications
  MikroTik router to MikroTik router
  MikroTik router to MikroTik router P2P using X.21 line
  MikroTik router to Cisco router using X.21 line
  MikroTik router to MikroTik router using Frame Relay

General Information

Summary

The MikroTik RouterOS supports FarSync T-Series X.21 synchronous adapter hardware. These cards provide versatile high performance connectivity to the Internet or to corporate networks over leased lines.

Specifications

Packages required: ***synchronous***
License required: ***level4***
Home menu level: ***/interface farsync***
Standards and Technologies: ***X.21, Frame Relay, PPP***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***
- ***Log Management***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Additional Documents

- ***http://www.farsite.co.uk/***

Synchronous Interface Configuration

Home menu level: ***/interface farsync***

Description

You can change the interface name to a more descriptive one using the **set** command. To enable the interface, use the **enable** command.

Property Description

hdlc-keepalive (*time*; default: **10s**) - Cisco HDLC keepalive period in seconds

clock-rate (*integer*; default: **64000**) - the speed of internal clock

clock-source (*external* | *internal*; default: **external**) - clock source

disabled (*yes* | *no*; default: **yes**) - shows whether the interface is disabled

frame-relay-dce (*yes* | *no*; default: **no**) - operate in Data Communications Equipment mode

frame-relay-lmi-type (*ansi* | *ccitt*; default: **ansi**) - Frame Relay Local Management Interface type

line-protocol (*cisco-hdlc* | *frame-relay* | *sync-ppp*; default: **sync-ppp**) - line protocol

media-type (*V24* | *V35* | *X21*; default: **V35**) - type of the media

mtu (*integer*; default: **1500**) - Maximum Transmit Unit

name (*name*; default: **farsyncN**) - assigned interface name

Example

```
[admin@MikroTik] > interface print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                TYPE             MTU
  0  R ether1               ether            1500
  1 X  farsync1             farsync          1500
  2 X  farsync2             farsync          1500
[admin@MikroTik] interface>
[admin@MikroTik] interface> enable 1
[admin@MikroTik] interface> enable farsync2
[admin@MikroTik] > interface print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                TYPE             MTU
  0  R ether1               ether            1500
  1    farsync1             farsync          1500
  2    farsync2             farsync          1500
[admin@MikroTik] interface>farsync
[admin@MikroTik] interface farsync> print
Flags: X - disabled, R - running
  0    name="farsync1" mtu=1500 line-protocol=sync-ppp media-type=V35
       clock-rate=64000 clock-source=external chdlc-keepalive=10s
       frame-relay-lmi-type=ansi frame-relay-dce=no

  1    name="farsync2" mtu=1500 line-protocol=sync-ppp media-type=V35
       clock-rate=64000 clock-source=external chdlc-keepalive=10s
       frame-relay-lmi-type=ansi frame-relay-dce=no
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] interface farsync>
```

You can monitor the status of the synchronous interface:

```
[admin@MikroTik] interface farsync> monitor 0
           card-type: T2P FarSync T-Series
               state: running
         firmware-id: 2
    firmware-version: 0.7.0
      physical-media: V35
               cable: detected
               clock: not-detected
       input-signals: CTS
      output-signals: RTS DTR

[admin@MikroTik] interface farsync>
```

Troubleshooting

Description

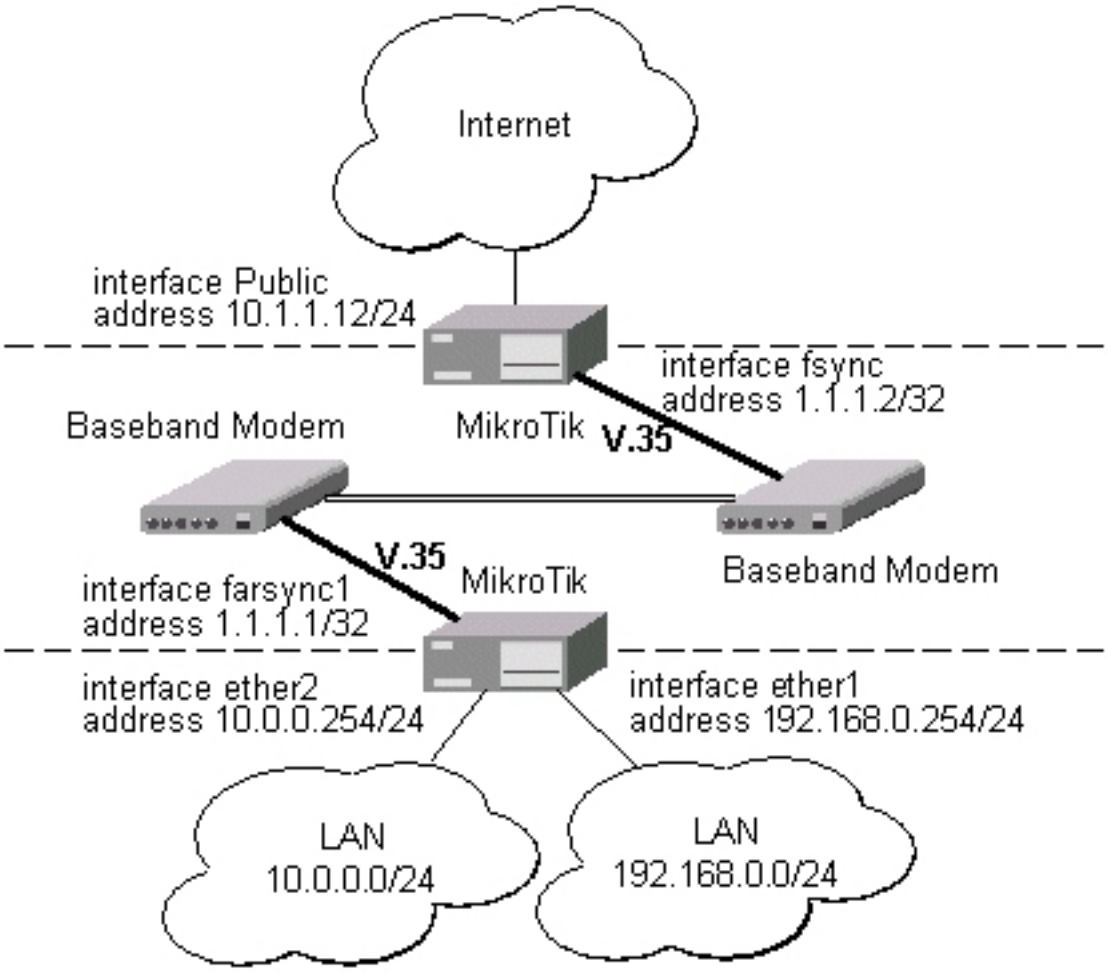
- **The farsync interface does not show up under the interface list**
  Obtain the required license for synchronous feature
- **The synchronous link does not work**
  Check the cabling and the line between the modems. Read the modem manual

Synchronous Link Applications

MikroTik router to MikroTik router

Let us consider the following network setup with two MikroTik routers connected to a leased line with baseband modems:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



The interface should be enabled according to the instructions given above. The **IP addresses** assigned to the synchronous interface should be as follows:

```
[admin@MikroTik] ip address> add address 1.1.1.1/32 interface farsync1 \
\... network 1.1.1.2 broadcast 255.255.255.255
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.254/24      10.0.0.254      10.0.0.255      ether2
  1   192.168.0.254/24   192.168.0.254   192.168.0.255   ether1
  2   1.1.1.1/32         1.1.1.2         255.255.255.255 farsync1
[admin@MikroTik] ip address> /ping 1.1.1.2
1.1.1.2 64 byte pong: ttl=255 time=31 ms
1.1.1.2 64 byte pong: ttl=255 time=26 ms
1.1.1.2 64 byte pong: ttl=255 time=26 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 26/27.6/31 ms
[admin@MikroTik] ip address>
```

Note that for the point-to-point link the network mask is set to 32 bits, the argument **network** is set to the **IP address** of the other end, and the broadcast address is set to 255.255.255.255. The default route should be set to the gateway router 1.1.1.2:

```
[admin@MikroTik] ip route> add gateway 1.1.1.2
[admin@MikroTik] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0  S 0.0.0.0/0          r 1.1.1.2         1        farsync1
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

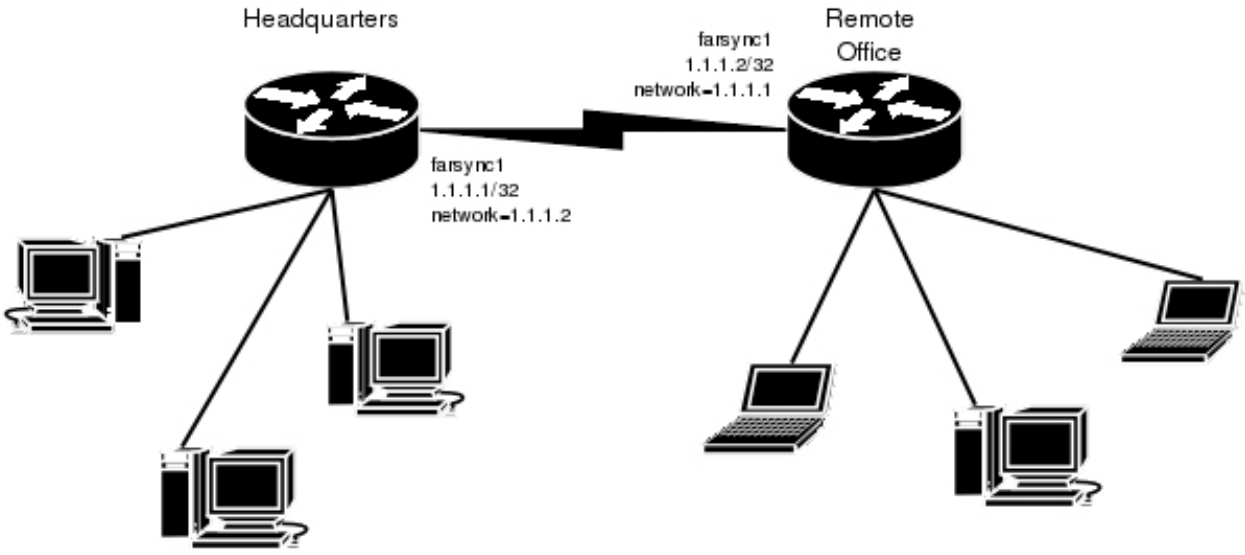
```
    1 DC 10.0.0.0/24        r 10.0.0.254      1        ether2
    2 DC 192.168.0.0/24     r 192.168.0.254   0        ether1
    3 DC 1.1.1.2/32         r 0.0.0.0         0        farsync1

[admin@MikroTik] ip route>
```

The configuration of the MikroTik router at the other end is similar:

```
[admin@MikroTik] ip address> add address 1.1.1.2/32 interface fsync \
\... network 1.1.1.1 broadcast 255.255.255.255
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.1.1.12/24       10.1.1.12       10.1.1.255      Public
  1   1.1.1.2/32         1.1.1.1         255.255.255.255 fsync
[admin@MikroTik] ip address> /ping 1.1.1.1
1.1.1.1 64 byte pong: ttl=255 time=31 ms
1.1.1.1 64 byte pong: ttl=255 time=26 ms
1.1.1.1 64 byte pong: ttl=255 time=26 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 26/27.6/31 ms
[admin@MikroTik] ip address>
```

MikroTik router to MikroTik router P2P using X.21 line

Consider the following example:



The default value of the property **clock-source** must be changed to **internal** for one of the cards. Both cards must have **media-type** property set to **X21**.

IP address configuration on both routers is as follows (by convention, the routers are named **hq** and **office** respectively):

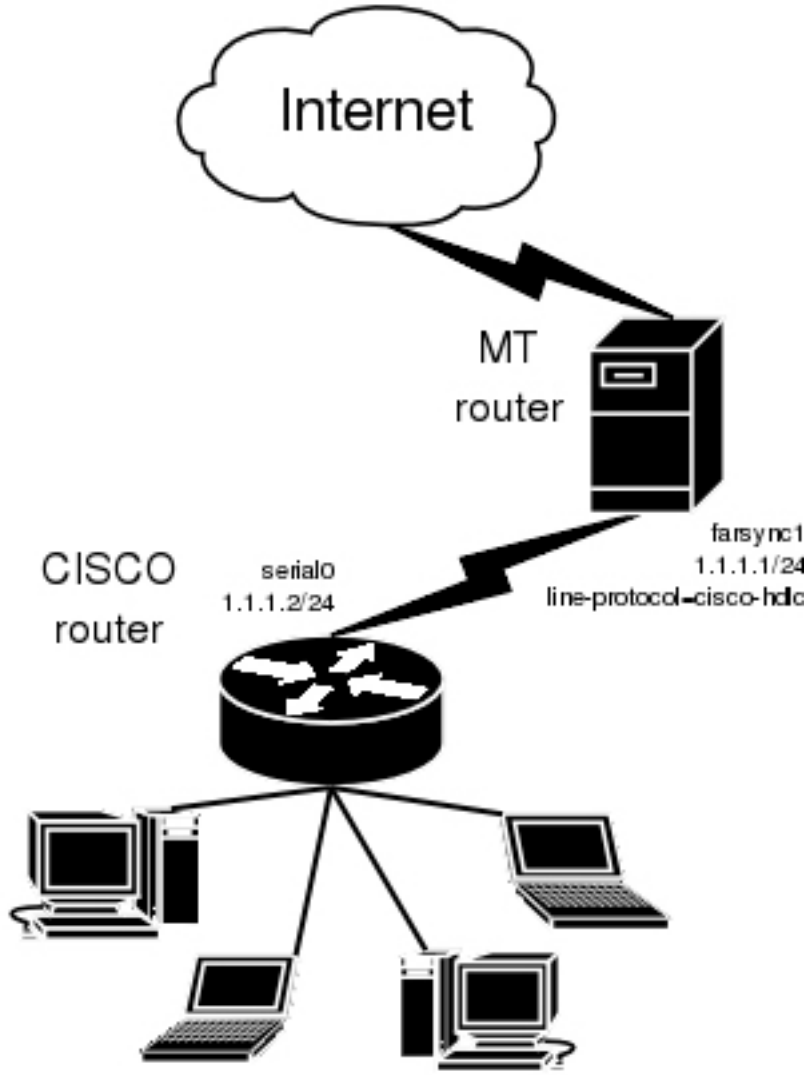
```
[admin@hq] ip address> pri
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   192.168.0.1/24     192.168.0.0     192.168.0.255   ether1
  1   1.1.1.1/32         1.1.1.2         1.1.1.2         farsync1

[admin@hq] ip address>

[admin@office] ip address>
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.112/24      10.0.0.0        10.0.0.255      ether1
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  1   1.1.1.2/32        1.1.1.1         1.1.1.1         farsync1

[admin@office] ip address>
```

MikroTik router to Cisco router using X.21 line

Assume we have the following configuration:



The configuration of MT router is as follows:

```
[admin@MikroTik] interface farsync> set farsync1 line-protocol=cisco-hdlc \
\... media-type=X21 clock-source=internal
[admin@MikroTik] interface farsync> enable farsync1
[admin@MikroTik] interface farsync> print
Flags: X - disabled, R - running
  0  R name="farsync1" mtu=1500 line-protocol=cisco-hdlc media-type=X21
       clock-rate=64000 clock-source=internal chdlc-keepalive=10s
       frame-relay-lmi-type=ansi frame-relay-dce=no

  1 X  name="farsync2" mtu=1500 line-protocol=sync-ppp media-type=V35
       clock-rate=64000 clock-source=external chdlc-keepalive=10s
       frame-relay-lmi-type=ansi frame-relay-dce=no

[admin@MikroTik] interface farsync>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

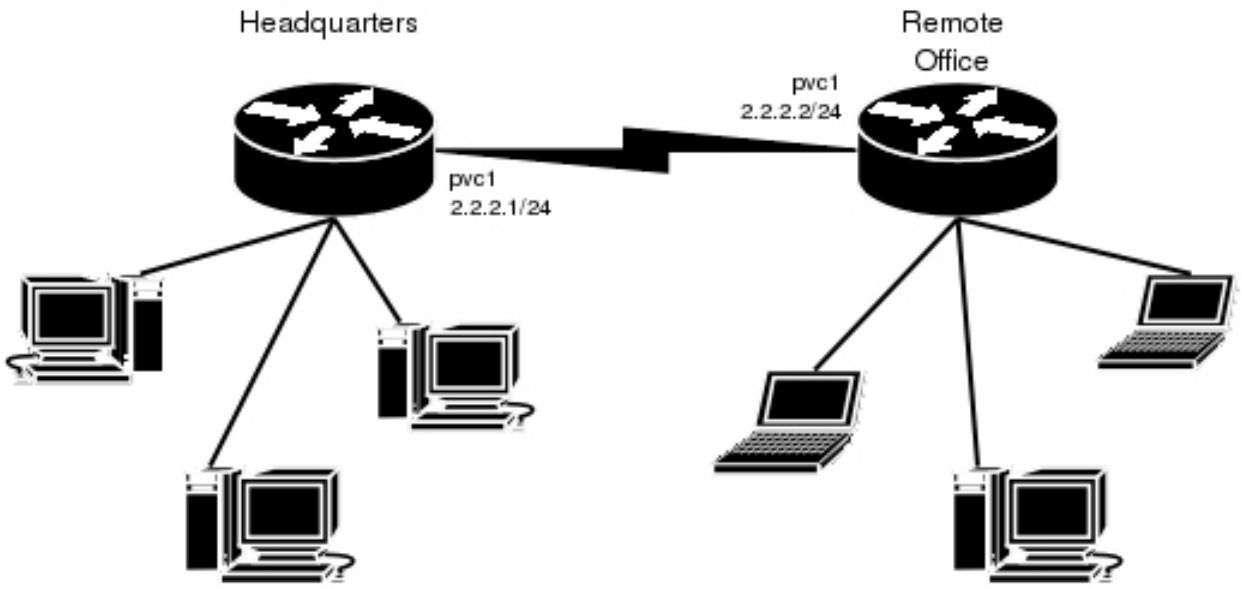
```
[admin@MikroTik] interface farsync> /ip address add address=1.1.1.1/24 \
\... interface=farsync1
```

The essential part of the configuration of Cisco router is provided below:

```
interface Serial0
 ip address 1.1.1.2 255.255.255.0
 no ip route-cache
 no ip mroute-cache
 no fair-queue
!
ip classless
ip route 0.0.0.0 0.0.0.0 1.1.1.1
```

MikroTik router to MikroTik router using Frame Relay

Consider the following example:



The default value of the property **clock-source** must be changed to **internal** for one of the cards. This card also requires the property **frame-relay-dce** set to **yes**. Both cards must have **media-type** property set to **X21** and the **line-protocol** set to **frame-relay**.

Now we need to add **pvc** interfaces:

```
[admin@hq] interface pvc> add dlci=42 interface=farsync1
[admin@hq] interface pvc> print
Flags: X - disabled, R - running
  #    NAME                                                MTU  DLCI INTERFACE
  0 X  pvc1                                                1500 42   farsync1

[admin@hq] interface pvc>
```

Similar routine has to be done also on **office** router:

```
[admin@office] interface pvc> add dlci=42 interface=farsync1
[admin@office] interface pvc> print
Flags: X - disabled, R - running
  #    NAME                                                MTU  DLCI INTERFACE
  0 X  pvc1                                                1500 42   farsync1

[admin@office] interface pvc>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Finally we need to add **IP addresses** to **pvc** interfaces and enable them.

On the **hq** router:

```
[admin@hq] interface pvc> /ip addr add address 2.2.2.1/24 interface pvc1
[admin@hq] interface pvc> /ip addr print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.112/24      10.0.0.0        10.0.0.255      ether1
  1   192.168.0.1/24     192.168.0.0     192.168.0.255   ether2
  2   2.2.2.1/24         2.2.2.0         2.2.2.255       pvc1

[admin@hq] interface pvc> enable 0
[admin@hq] interface pvc>
```

and on the **office** router:

```
[admin@office] interface pvc> /ip addr add address 2.2.2.2/24 interface pvc1
[admin@office] interface pvc> /ip addr print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.112/24      10.0.0.0        10.0.0.255      ether1
  1   2.2.2.2/24         2.2.2.0         2.2.2.255       pvc1

[admin@office] interface pvc> enable 0
[admin@office] interface pvc>
```

Now we can monitor the synchronous link status:

```
[admin@hq] interface pvc> /ping 2.2.2.2
2.2.2.2 64 byte ping: ttl=64 time=20 ms
2.2.2.2 64 byte ping: ttl=64 time=20 ms
2.2.2.2 64 byte ping: ttl=64 time=21 ms
2.2.2.2 64 byte ping: ttl=64 time=21 ms
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 20/20.5/21 ms
[admin@hq] interface pvc> /interface farsync monitor 0
           card-type: T2P FarSync T-Series
               state: running-normally
         firmware-id: 2
    firmware-version: 1.0.1
            physical: X.21
               cable: detected
               clock: detected
       input-signals: CTS
      output-signals: RTS,DTR

[admin@hq] interface pvc>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

FrameRelay (PVC, Private Virtual Circuit) Interface

Document revision 1.1 (Fri Mar 05 08:14:41 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Description
  Additional Documents
Configuring Frame Relay Interface
  Description
  Property Description
  Notes
Frame Relay Configuration
  Example with Cyclades Interface
  Example with MOXA Interface
  Example with MikroTik Router to MikroTik Router
Troubleshooting
  Description

General Information

Summary

Frame Relay is a multiplexed interface to packet switched network and is a simplified form of Packet Switching similar in principle to X.25 in which synchronous frames of data are routed to different destinations depending on header information. Frame Relay uses the synchronous HDLC frame format.

Specifications

Packages required: ***synchronous***
License required: ***level4***
Home menu level: ***/interface pvc***
Standards and Technologies: ***Frame Relay (RFC1490)***
Hardware usage: ***Not significant***

Description

To use Frame Relay interface you must have already working synchronous interface. You can read how to set up synchronous boards supported by MikroTik RouterOS:

- ***Cyclades PC300 PCI Adapters***
- ***Moxa C101 Synchronous interface***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***Moxa C502 Dual Port Synchronous interface***

Additional Documents

- ***Frame Relay Forum***
- ***http://www2.rad.com/networks/1994/fram_rel/frame.htm***

Configuring Frame Relay Interface

Home menu level: ***/interface pvc***

Description

To configure frame relay, at first you should set up the synchronous interface, and then the PVC interface.

Property Description

name (*name*; default: **pvcN**) - assigned name of the interface

mtu (*integer*; default: **1500**) - Maximum Transmission Unit of an interface

dlci (*integer*; default: **16**) - Data Link Connection Identifier assigned to the PVC interface

interface (*name*) - Frame Relay interface

Notes

A DLCI is a channel number (Data Link Connection Identifier) which is attached to data frames to tell the network how to route the data. Frame Relay is "statistically multiplexed", which means that only one frame can be transmitted at a time but many logical connections can co-exist on a single physical line. The DLCI allows the data to be logically tied to one of the connections so that once it gets to the network, it knows where to send it.

Frame Relay Configuration

Example with Cyclades Interface

Let us consider the following network setup with MikroTik router with Cyclades PC300 interface connected to a leased line with baseband modems and a Cisco router at the other end.

```
[admin@MikroTik] ip address> add interface=pvc1 address=1.1.1.1 netmask=255.255.255.0
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   1.1.1.1/24         1.1.1.0         1.1.1.255       pvc1
[admin@MikroTik] ip address>
```

PVC and Cyclades interface configuration

- Cyclades

```
[admin@MikroTik] interface cyclades> print
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
Flags: X - disabled, R - running
  0  R name="cyclades1" mtu=1500 line-protocol=frame-relay media-type=V35
       clock-rate=64000 clock-source=external line-code=B8ZS framing-mode=ESF
       line-build-out=0dB rx-sensitivity=short-haul frame-relay-lmi-type=ansi
       frame-relay-dce=no chdlc-keepalive=10s

[admin@MikroTik] interface cyclades>
```

- PVC

```
[admin@MikroTik] interface pvc> print
Flags: X - disabled, R - running
  #    NAME                MTU  DLCI INTERFACE
  0  R pvc1                1500 42   cyclades1
[admin@MikroTik] interface pvc>
```

- Cisco router setup

```
CISCO# show running-config

Building configuration...

Current configuration...

...
!
ip subnet-zero
no ip domain-lookup
frame-relay switching
!
interface Ethernet0
 description connected to EthernetLAN
 ip address 10.0.0.254 255.255.255.0
!
interface Serial0
 description connected to Internet
 no ip address
 encapsulation frame-relay IETF
 serial restart-delay 1
 frame-relay lmi-type ansi
 frame-relay intf-type dce
!
interface Serial0.1 point-to-point
 ip address 1.1.1.2 255.255.255.0
 no arp frame-relay
 frame-relay interface-dlci 42
!
...
end.
```

Send ping to MikroTik router

```
CISCO#ping 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/31/32 ms
CISCO#
```

Example with MOXA Interface

Let us consider the following network setup with MikroTik router with MOXA C502 synchronous interface connected to a leased line with baseband modems and a Cisco router at the other end.

```
[admin@MikroTik] ip address> add interface=pvc1 address=1.1.1.1 netmask=255.255.255.0
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  0   1.1.1.1/24         1.1.1.0         1.1.1.255       pvc1
[admin@MikroTik] ip address>
```

PVC and Moxa interface configuration

- Moxa

```
[admin@MikroTik] interface moxa-c502> print
Flags: X - disabled, R - running
  0  R name="moxa1" mtu=1500 line-protocol=frame-relay clock-rate=64000
       clock-source=external frame-relay-lmi-type=ansi frame-relay-dce=no
       cisco-hdlc-keepalive-interval=10s

  1 X  name="moxa-c502-2" mtu=1500 line-protocol=sync-ppp clock-rate=64000
       clock-source=external frame-relay-lmi-type=ansi frame-relay-dce=no
       cisco-hdlc-keepalive-interval=10s

[admin@MikroTik] interface moxa-c502>
```

- PVC

```
[admin@MikroTik] interface pvc> print
Flags: X - disabled, R - running
  #    NAME                MTU  DLCI INTERFACE
  0  R pvc1                1500 42   moxa1
[admin@MikroTik] interface pvc>

CISCO router setup

CISCO# show running-config

Building configuration...

Current configuration...

...
!
ip subnet-zero
no ip domain-lookup
frame-relay switching
!
interface Ethernet0
 description connected to EthernetLAN
 ip address 10.0.0.254 255.255.255.0
!
interface Serial0
 description connected to Internet
 no ip address
 encapsulation frame-relay IETF
 serial restart-delay 1
 frame-relay lmi-type ansi
 frame-relay intf-type dce
!
interface Serial0.1 point-to-point
 ip address 1.1.1.2 255.255.255.0
 no arp frame-relay
 frame-relay interface-dlci 42
!
...
end.

Send ping to MikroTik router

CISCO#ping 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/31/32 ms
CISCO#
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example with MikroTik Router to MikroTik Router

Let us consider the following example:

In this example we will use two Moxa C101 synchronous cards.

Do not forget to set **line-protocol** for synchronous interfaces to **frame-relay**. To achieve proper result, one of the synchronous interfaces must operate in DCE mode:

```
[admin@r1] interface moxa-c101> set 0 frame-relay-dce=yes
[admin@r1] interface moxa-c101> print
Flags: X - disabled, R - running
  0  R name="moxa-c101-1" mtu=1500 line-protocol=frame-relay clock-rate=64000
       clock-source=external frame-relay-lmi-type=ansi frame-relay-dce=yes
       cisco-hdlc-keepalive-interval=10s ignore-dcd=no

[admin@r1] interface moxa-c101>
```

Then we need to add PVC interfaces and **IP addresses**.

On the **R1**:

```
[admin@r1] interface pvc> add dlci=42 interface=moxa-c101-1
[admin@r1] interface pvc> print
Flags: X - disabled, R - running
  #    NAME                                                MTU  DLCI INTERFACE
  0 X  pvc1                                                1500 42   moxa-c101-1

[admin@r1] interface pvc> /ip address add address 4.4.4.1/24 interface pvc1
```

on the **R2**:

```
[admin@r2] interface pvc> add dlci=42 interface=moxa-c101-1
[admin@r2] interface pvc> print
Flags: X - disabled, R - running
  #    NAME                                                MTU  DLCI INTERFACE
  0 X  pvc1                                                1500 42   moxa-c101-1

[admin@r2] interface pvc> /ip address add address 4.4.4.2/24 interface pvc1
```

Finally, we must enable PVC interfaces:

```
[admin@r1] interface pvc> enable pvc1
[admin@r1] interface pvc>

[admin@r2] interface pvc> enable pvc1
[admin@r2] interface pvc>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Troubleshooting

Description

- **I cannot ping through the synchronous frame relay interface between MikroTik router and a Cisco router**
  Frame Relay does not support address resolving and IETF encapsulation should be used. Please check the configuration on the Cisco router

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

GPRS PCMCIA

Document revision 1.0 (Fri Jul 15 15:07:41 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
How to make a GPRS connection
  Description
  Example

How to make a GPRS connection

Description

Let us consider a situation that you are in a place where no internet connection is available, but you have access to your mobile network provider. In this case you can connect MikroTik router to your mobile phone provider using GPRS (General Packet Radio Service) and so establish an internet connection.

In this example we are using a ***PCMCIA GPRS card***.

Example

- Plug the GPRS PCMCIA card (with your SIM card) into the router, turn on the router and after it has started, see if a new port has appeared. In this case it is the **serial1** port which is our GPRS device:

```
[admin@MikroTik] port> print
 # NAME                                USED-BY                             BAUD-RATE
 0 serial0                             Serial Console                      115200
 1 serial1                                                                 9600
[admin@MikroTik] port>
```

- Enter the pin code from serial-terminal (in this case, PIN code is **3663**) :

```
/system serial-terminal serial1

AT+CPIN=”3663”
```

  Now you should see **OK** on your screen. Wait for about 5 seconds and see if the green led started to blink. Press Ctrl+Q to quit the serial-terminal.

- Change remote-address in **/ppp profile**, in this case to 212.93.96.65 (you should obtain it from your mobile network operator):

```
/ppp profile set default remote-address=212.93.96.65
```

- Add a ppp client:

```
/interface ppp-client add dial-command=ATD phone=*99***1# \
\... modem-init="AT+CGDCONT=1,\"IP\",\"internet\"" port=serial1
```

- Now enable the interface and see if it is connected:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] interface ppp-client> enable 0
[admin@MikroTik] interface ppp-client> mo 0
    status: dialing...

    status: link established

       status: authenticated
       uptime: 0s
    idle-time: 0s

       status: authenticated
       uptime: 1s
    idle-time: 1s

       status: connected
       uptime: 2s
    idle-time: 2s
[admin@MikroTik] interface ppp-client>
```

Check the IP addresses:

```
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   192.168.0.5/24     192.168.0.0     192.168.0.255   ether1
 1 D 10.40.205.168/32   212.93.96.65    0.0.0.0         ppp-out1
[admin@MikroTik] ip address>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

ISDN (Integrated Services Digital Network) Interface

Document revision 1.1 (Fri Mar 05 08:15:11 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Additional Documents
ISDN Hardware and Software Installation
  Description
  Property Description
  ISDN Channels
  MSN and EAZ numbers
ISDN Client Interface Configuration
  Description
  Property Description
  Example
ISDN Server Interface Configuration
  Description
  Property Description
  Example
ISDN Examples
  ISDN Dial-out
  ISDN Dial-in
  ISDN Backup

General Information

Summary

The MikroTik router can act as an ISDN client for dialing out, or as an ISDN server for accepting incoming calls. The dial-out connections may be set as dial-on-demand or as permanent connections (simulating a leased line). The remote **IP address** (provided by the ISP) can be used as the default gateway for the router.

Specifications

Packages required: ***isdn, ppp***
License required: ***level1***
Home menu level: ***/interface isdn-server, /interface isdn-client***
Standards and Technologies: ***PPP (RFC 1661)***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***Log Management***

Additional Documents

- ***PPP over ISDN***
- ***RFC3057 - ISDN Q.921-User Adaptation Layer***

ISDN Hardware and Software Installation

Command name: ***/driver add***

Description

Please install the ISDN adapter into the PC accordingly the instructions provided by the adapter manufacturer.

Appropriate packages have to be downloaded from MikroTik??????s web page ***http://www.mikrotik.com***. After all, the ISDN driver should be loaded using the **/driver add** command.

MikroTik RouterOS supports passive PCI adapters with Siemens chipset:

- Eicon. Diehl Diva - **diva**
- Sedlbauer Speed - **sedlbauer**
- ELSA Quickstep 1000 - **quickstep**
- NETjet - **netjet**
- Teles - **teles**
- Dr. Neuhaus Niccy - **niccy**
- AVM - **avm**
- Gazel - **gazel**
- HFC 2BDS0 based adapters - **hfc**
- W6692 based adapters - **w6692**

For example, for the HFC based PCI card, it is enough to use **/driver add name=hfc** command to get the driver loaded.

Note! ISDN **ISA** adapters are **not** supported!

Property Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

name (*name*) - name of the driver

isdn-protocol (*euro* | *german*; default: **euro**) - data channel protocol

ISDN Channels

ISDN channels are added to the system automatically when the ISDN card driver is loaded. Each channel corresponds to one physical 64K ISDN data channel.

The list of available ISDN channels can be viewed using the **/isdn-channels print** command. The channels are named **channel1**, **channel2**, and so on. E.g., if you have two ISDN channels, and one of them currently used by an ISDN interface, but the other available, the output should look like this:

```
[admin@MikroTik] isdn-channels> print
Flags: X - disabled, E - exclusive
  #    NAME                 CHANNEL    DIR.. TYPE  PHONE
  0    channel1             0
  1    channel2             1
[admin@MikroTik] isdn-channels>
```

ISDN channels are very similar to PPP serial ports. Any number of ISDN interfaces can be configured on a single channel, but only one interface can be enabled for that channel at a time. It means that every ISDN channel is either available or used by an ISDN interface.

MSN and EAZ numbers

In Euro-ISDN a subscriber can assign more than one ISDN number to an ISDN line. For example, an ISDN line could have the numbers 1234067 and 1234068. Each of these numbers can be used to dial the ISDN line. These numbers are referred to as Multiple Subscriber Numbers (MSN).

A similar, but separate concept is EAZ numbering, which is used in German ISDN networking. EAZ number can be used in addition to dialed phone number to specify the required service.

For dial-out ISDN interfaces, MSN/EAZ number specifies the outgoing phone number (the calling end). For dial-in ISDN interfaces, MSN/EAZ number specifies the phone number that will be answered. If you are unsure about your MSN/EAZ numbers, leave them blank (it is the default).

For example, if your ISDN line has numbers 1234067 and 1234068, you could configure your dial-in server to answer only calls to 1234068 by specifying **1234068** as your MSN number. In a sense, MSN is just your phone number.

ISDN Client Interface Configuration

Home menu level: ***/interface isdn-client***

Description

The ISDN client is used to connect to remote dial-in server (probably ISP) via ISDN. To set up an ISDN dial-out connection, use the ISDN dial-out configuration menu under the submenu.

Property Description

name (*name*; default: **isdn-outN**) - interface name

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

mtu (*integer*; default: **1500**) - Maximum Transmission Unit

mru (*integer*; default: **1500**) - Maximum Receive Unit

phone (*integer*; default: **""**) - phone number to dial

msn (*integer*; default: **""**) - MSN/EAZ of ISDN line provided by the line operator

dial-on-demand (*yes* | *no*; default: **no**) - use dialing on demand

l2-protocol (*hdlc* | *x75i* | *x75ui* | *x75bui*; default: **hdlc**) - level 2 protocol to be used

user (*text*) - user name that will be provided to the remote server

password (*text*) - password that will be provided to the remote server

allow (*multiple choice: mschap2, mschap1, chap, pap*; default: **mschap2, mschap1, chap, pap**) - the protocol to allow the client to use for authentication

add-default-route (*yes* | *no*; default: **no**) - add default route to remote host on connect

profile (*name*; default: **default**) - profile to use when connecting to the remote server

use-peer-dns (*yes* | *no*; default: **no**) - use or not peer DNS

bundle-128K (*yes* | *no*; default: **yes**) - use both channels instead of just one

Example

ISDN client interfaces can be added using the **add** command:

```
[admin@MikroTik] interface isdn-client> add msn="142" user="test" \
\... password="test" phone="144" bundle-128K=no
[admin@MikroTik] interface isdn-client> print
Flags: X - disabled, R - running
  0 X  name="isdn-out1" mtu=1500 mru=1500 msn="142" user="test"
       password="test" profile=default phone="144" l2-protocol=hdlc
       bundle-128K=no dial-on-demand=no add-default-route=no use-peer-dns=no

[admin@MikroTik] interface isdn-client>
```

ISDN Server Interface Configuration

Home menu level: ***/interface isdn-client***

Description

ISDN server is used to accept remote dial-in connections form ISDN clients.

Property Description

name (*name*; default: **isdn-inN**) - interface name

mtu (*integer*; default: **1500**) - Maximum Transmission Unit

mru (*integer*; default: **1500**) - Maximum Receive Unit

phone (*integer*; default: **""**) - phone number to dial

msn (*integer*; default: **""**) - MSN/EAZ of ISDN line provided by the line operator

l2-protocol (*hdlc* | *x75i* | *x75ui* | *x75bui*; default: **hdlc**) - level 2 protocol to be used

profile (*name*; default: **default**) - profile to use when connecting to the remote server

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

bundle-128K (*yes* | *no*; default: **yes**) - use both channels instead of just one

authentication (*pap* | *chap* | *mschap1* | *mschap2*; default: **mschap2, mschap1, chap, pap**) - used authentication

Example

ISDN server interfaces can be added using the **add** command:

```
[admin@MikroTik] interface isdn-server> add msn="142" bundle-128K=no
[admin@MikroTik] interface isdn-server> print
Flags: X - disabled, R - running
  0 X  name="isdn-in1" mtu=1500 mru=1500 msn="142"
       authentication=mschap2,chap,pap profile=default l2-protocol=x75bui
       bundle-128K=no

[admin@MikroTik] interface isdn-server>
```

ISDN Examples

ISDN Dial-out

Dial-out ISDN connections allow a local router to connect to a remote dial-in server (ISP's) via ISDN.

Let's assume you would like to set up a router that connects your local LAN with your ISP via ISDN line. First you should load the corresponding ISDN card driver. Supposing you have an ISDN card with a **W6692**-based chip:

```
[admin@MikroTik]> /driver add name=w6692
```

Now additional channels should appear. Assuming you have only one ISDN card driver loaded, you should get following:

```
[admin@MikroTik] isdn-channels> print
Flags: X - disabled, E - exclusive
  #    NAME                     CHANNEL    DIR.. TYPE  PHONE
  0    channel1                 0
  1    channel2                 1
[admin@MikroTik] isdn-channels>
```

Suppose you would like to use dial-on-demand to dial your ISP and automatically add a default route to it. Also, you would like to disconnect when there is more than 30s of network inactivity. Your ISP's phone number is 12345678 and the user name for authentication is 'john'. Your ISP assigns IP addresses automatically. Add an outgoing ISDN interface and configure it in the following way:

```
[admin@mikrotik]> /interface isdn-client add name="isdn-isp" phone="12345678"
user="john" password="31337!)" add-default-route=yes dial-on-demand=yes
[admin@MikroTik] > /interface isdn-client print
Flags: X - disabled, R - running
  0 X  name="isdn-isp" mtu=1500 mru=1500 msn="" user="john" password="31337!)"
       profile=default phone="12345678" l2-protocol=hdlc bundle-128K=no
       dial-on-demand=yes add-default-route=yes use-peer-dns=no
```

Configure PPP profile.

```
[admin@MikroTik] ppp profile> print
Flags: * - default
  0 * name="default" local-address=0.0.0.0 remote-address=0.0.0.0
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
      session-timeout=0s idle-timeout=0s use-compression=no
      use-vj-compression=yes use-encryption=no require-encryption=no only-one=no
      tx-bit-rate=0 rx-bit-rate=0 incoming-filter="" outgoing-filter=""

[admin@Mikrotik] ppp profile> set default idle-timeout=30s
```

If you would like to remain connected all the time, i.e., as a leased line, then set the **idle-timeout** to 0s.

All that remains is to enable the interface:

```
[admin@MikroTik] /interface set isdn-isp disabled=no
```

You can monitor the connection status with the following command:

```
[admin@MikroTik] /interface isdn-client monitor isdn-isp
```

ISDN Dial-in

Dial-in ISDN connections allow remote clients to connect to your router via ISDN.

Let us assume you would like to configure a router for accepting incoming ISDN calls from remote clients. You have an Ethernet card connected to the LAN, and an ISDN card connected to the ISDN line. First you should load the corresponding ISDN card driver. Supposing you have an ISDN card with an HFC chip:

```
[admin@MikroTik] /driver add name=hfc
```

Now additional channels should appear. Assuming you have only one ISDN card driver loaded, you should get the following:

```
[admin@MikroTik] isdn-channels> print
Flags: X - disabled, E - exclusive
  #    NAME                      CHANNEL    DIR.. TYPE  PHONE
  0    channel1                  0
  1    channel2                  1
[admin@MikroTik] isdn-channels>
```

Add an incoming ISDN interface and configure it in the following way:

```
[admin@MikroTik] interface isdn-server> add msn="7542159" \
\... authentication=chap,pap bundle-128K=no
[admin@MikroTik] interface isdn-server> print
Flags: X - disabled
  0 X  name="isdn-in1" mtu=1500 mru=1500 msn="7542159" authentication=chap,pap
       profile=default l2-protocol=hldc bundle-128K=no
```

Configure PPP settings and add users to router's database.

```
[admin@MikroTik] ppp profile> print
Flags: * - default
  0 * name="default" local-address=0.0.0.0 remote-address=0.0.0.0
      session-timeout=0s idle-timeout=0s use-compression=no
      use-vj-compression=yes use-encryption=no require-encryption=no only-one=no
      tx-bit-rate=0 rx-bit-rate=0 incoming-filter="" outgoing-filter=""
[admin@Mikrotik] ppp profile> set default idle-timeout=5s local-address=10.99.8.1 \
\... remote-address=10.9.88.1
```

Add user 'john' to the router's user database. Assuming that the password is '31337!)':

```
[admin@MikroTik] ppp secret> add name=john password="31337!)" service=isdn
[admin@MikroTik] ppp secret> print
[admin@ISDN] ppp secret> print
Flags: X - disabled
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  #   NAME             SERVICE CALLER-ID        PASSWORD         PROFILE
  0   john             isdn                     31337!)          default
[admin@MikroTik] ppp secret>
```

Check the status of the ISDN server interface and wait for the call:

```
[admin@MikroTik] interface isdn-server> monitor isdn-in1

    status: Waiting for call...
```

ISDN Backup

Backup systems are used in specific cases, when you need to maintain a connection, even if a fault occurs. For example, if someone cuts the wires, the router can automatically connect to a different interface to continue its work. Such a backup is based on an utility that monitors the status of the connection - netwatch, and a script, which runs the netwatch.

This is an example of how to make simple router backup system. In this example we'll use an ISDN connection for purpose to backup a standard Ethernet connection. You can, however, use instead of the ISDN connection anything you need - PPP, for example. When the Ethernet fail (the router nr.1 cannot ping the router nr.2 to 2.2.2.2 (see picture) the router nr.1 will establish an ISDN connection, so-called backup link, to continue communicating with the nr. 2.

You must keep in mind, that in our case there are just two routers, but this system can be extended to support more different networks.

The backup system example is shown in the following picture:

In this case the **backup** interface is an ISDN connection, but in real applications it can be substituted by a particular connection. Follow the instructions below on how to set up the backup link:

- At first, you need to set up ISDN connection. To use ISDN, the ISDN card driver must be loaded:

```
[admin@MikroTik] driver> add name=hfc
```

  The PPP connection must have a new user added to the routers one and two:

```
[admin@Mikrotik] ppp secret> add name=backup password=backup service=isdn
```

  An ISDN server and PPP profile must be set up on the second router:

```
[admin@MikroTik] ppp profile> set default local-address=3.3.3.254
remote-address=3.3.3.1
[admin@MikroTik] interface isdn-server> add name=backup msn=7801032
```

  An ISDN client must be added to the first router:

```
[admin@MikroTik] interface isdn-client>
add name=backup user="backup" password="backup" phone=7801032 msn=7542159
```

- Then, you have to set up static routes
  Use the **/ip route add** command to add the required static routes and comments to them. Comments are required for references in scripts.
  The **first** router:

```
[admin@Mikrotik] ip route> add gateway 2.2.2.2 comment "route1"
```

  The **second** router:

```
[admin@Mikrotik] ip route> add gateway 2.2.2.1 comment "route1" dst-address 1.1.1.0/24
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- And finally, you have to add scripts.
  Add scripts in the submenu **/system script** using the following commands:
  The **first** router:

```
[admin@Mikrotik] system script> add name=connection_down \
\... source={/interface enable backup; /ip route set route1 gateway 3.3.3.254}
[admin@Mikrotik] system script> add name=connection_up \
\... source={/interface disable backup; /ip route set route1 gateway 2.2.2.2}
```

  The **second** router:

```
[admin@Mikrotik] system script> add name=connection_down \
\... source={/ip route set route1 gateway 3.3.3.1}
[admin@Mikrotik] system script> add name=connection_up \
\... source={/ip route set route1 gateway 2.2.2.1}
```

- To get all above listed to work, set up Netwatch utility. To use netwatch, you need the advanced tools feature package installed. Please upload it to the router and reboot. When installed, the advanced-tools package should be listed under the **/system package print** list.
  Add the following settings to the first router:

```
[admin@Mikrotik] tool netwatch> add host=2.2.2.1 interval=5s \
\... up-script=connection_up down-script=connection_down
```

  Add the following settings to the second router:

```
[admin@Mikrotik] tool netwatch> add host=2.2.2.2 interval=5s \
\... up-script=connection_up down-script=connection_down
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

LMC/SBEI Synchronous Interfaces

Document revision 0.3 (Wed Oct 13 13:18:32 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
Synchronous Interface Configuration
  Description
  Property Description
  Connecting two MT routers via T1 crossover

General Information

Summary

The MikroTik RouterOS supports the following Lanmedia Corp (LMC)/SBE Inc interfaces:

- LMC/SBEI wanPCI-1T3 PCI T3 (also known as DS3, 44.736Mbps)
- LMC/SBEI wanPCI-1T1E1 PCI T1/E1 (also known as DS1 or LMC1200P, 1.544 Mbps or 2.048 Mbps)

Specifications

Packages required: ***synchronous***
License required: ***level4***
Home menu level: ***/interface sbe***
Standards and Technologies: ***T1/E1/T3/G.703, Frame Relay, PPP, Cisco-HDLC***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***
- ***Log Management***

Synchronous Interface Configuration

Home menu level: ***/interface sbe***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

With the introduction of 2.8 release, MikroTik RouterOS supports popular SBEI wanPCI-1T3 and wanPCI-1T1E1 cards. These cards provide a router with the ability to communicate over T1, E1 and T3 links directly, without the need of external CSU/DSU equipment.

Property Description

chdlc-keepalive (*time*; default: **10s**) - specifies the keepalive interval for Cisco HDLC protocol

circuit-type (*e1* | *e1-cas* | *e1-plain* | *e1-unframed* | *t1* | *t1-unframed*; default: **e1**) - the circuit type particular interface is connected to

clock-rate (*integer*; default: **64000**) - internal clock rate in bps

clock-source (*internal* | *external*; default: **external**) - specifies whether the card should rely on supplied clock or generate its own

crc32 (yes | no; default: **no**) - Specifies whether to use CRC32 error correction algorithm or not

frame-relay-dce (*yes* | *no*; default: **no**) - specifies whether the device operates in Data Communication Equipment mode. The value yes is suitable only for T1 models

frame-relay-lmi-type (*ansi* | *ccitt*; default: **ansi**) - Frame Relay Line Management Interface Protocol type

line-protocol (*cisco-hdlc* | *frame-relay* | *sync-ppp*; default: **sync-ppp**) - encapsulated line protocol

long-cable (yes | no; default: **no**) - specifies whether to use signal phase shift for very long links

mtu (*integer*: 68..1500; default: **1500**) - IP protocol Maximum Transmission Unit

name (*name*; default: **sbeN**) - unique interface name.

scrambler (yes | no; default: **no**) - when enabled, makes the card unintelligible to anyone without a special receiver

General Information

Connecting two MT routers via T1 crossover

In the following example we will configure two routers to talk to each other via T1 link. The routers are named R1 and R2 with the addresses of 10.10.10.1/24 and 10.10.10.2/24, respectively. Cisco HDLC will be used as incapsulation protocol and circuit type will be regular T1.

First, we need to configure synchronous interfaces on both routers. Keep in mind, that one of the interfaces needs to be set to use its internal clock.

- On **R1** router:

```
[admin@MikroTik] > /interface sbe set sbe1 line-protocol=cisco-hdlc \ \...
clock-source=internal circuit-type=t1 disabled=no [admin@R1] > /interface sbe print
Flags: X - disabled, R - running 0 R name="sbe1" mtu=1500 line-protocol=cisco-hdlc
clock-rate=64000 clock-source=internal crc32=no long-cable=no scrambler=no
circuit-type=t1 frame-relay-lmi-type=ansi frame-relay-dce=no chdlc-keepalive=10s
[admin@R1] >
```

- On **R2** router:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] > /interface sbe set sbe1 line-protocol=cisco-hdlc \ \...
circuit-type=t1 disabled=no [admin@R2] > /interface sbe print Flags: X - disabled, R
- running 0 R name="sbe1" mtu=1500 line-protocol=cisco-hdlc clock-rate=64000
clock-source=external crc32=no long-cable=no scrambler=no circuit-type=t1
frame-relay-lmi-type=ansi frame-relay-dce=no chdlc-keepalive=10s [admin@R2] >
```

Then, we should assign IP addresses to both interfaces.

- On **R1** router:
  ```
  [admin@R1] > /ip address add address 10.10.10.1/24 interface=sbe1
  ```
- On **R2** router:
  ```
  [admin@R1] > /ip address add address 10.10.10.2/24 interface=sbe1
  ```

Finally, we could test connection by issuing **ping** command from **R1** router:

```
[admin@R1] > /ping 10.10.10.2
10.10.10.2 64 byte ping: ttl=64 time=7 ms
10.10.10.2 64 byte ping: ttl=64 time=8 ms
10.10.10.2 64 byte ping: ttl=64 time=8 ms
10.10.10.2 64 byte ping: ttl=64 time=8 ms
10.10.10.2 64 byte ping: ttl=64 time=8 ms
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 7/7.8/8 ms
[admin@R2] >
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

M3P

Document revision 0.3.0 (Wed Mar 03 16:07:55 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
Setup
  Description
  Property Description
  Notes
  Example

General Information

Summary

The MikroTik Packet Packer Protocol (M3P) optimizes the data rate usage of links using protocols that have a high overhead per packet transmitted. The basic purpose of this protocol is to better enable wireless networks to transport VoIP traffic and other traffic that uses small packet sizes of around 100 bytes.

M3P features:

- enabled by a per interface setting
- other routers with MikroTik Discovery Protocol enabled will broadcast M3P settings
- significantly increases bandwidth availability over some wireless links ? by approximately four times
- offer configuration settings to customize this feature

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip packing***
Standards and Technologies: ***M3P***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***MNDP***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

The wireless protocol IEEE 802.11 and, to a lesser extent, Ethernet protocol have a high overhead per packet as for each packet it is necessary to access the media, check for errors, resend in case of errors occured, and send network maintenance messages (network maintenance is applicable only for wireless). The MikroTik Packet Packer Protocol improves network performance by aggregating many small packets into a big packet, thereby minimizing the network per packet overhead cost. The M3P is very effective when the average packet size is 50-300 bytes ? the common size of VoIP packets.

Features:

- may work on any Ethernet-like media
- is disabled by default for all interfaces
- when older version on the RouterOS are upgraded from a version without M3P to a version with discovery, current wireless interfaces will not be automatically enabled for M3P
- small packets going to the same MAC level destination (regardless of IP destination) are collected according to the set configuration and aggregated into a large packet according to the set size
- the packet is sent as soon as the maximum aggregated-packet packet size is reached or a maximum time of 15ms (+/-5ms)

Setup

Home menu level: ***/ip packing***

Description

M3P is working only between MikroTik routers, which are discovered with MikroTik Neighbor Discovery Protocol (MNDP). When M3P is enabled router needs to know which of its neighbouring hosts have enabled M3P. MNDP is used to negotiate unpacking settings of neighbours, therefore it has to be enabled on interfaces you wish to enable M3P. Consult MNDP manual on how to do it.

Property Description

aggregated-size (*integer*; default: **1500**) - the maximum aggregated packet's size

interface (*name*) - interface to enable M3P on

packing (*none* | *simple* | *compress-all* | *compress-headers*; default: **simple**) - specifies the packing mode

- **none** - no packing is applied to packets
- **simple** - aggregate many small packets into one large packet, minimizing network overhead per packet
- **compress-headers** - further increase network performance by compressing IP packet header (consumes more CPU resources)
- **compress-all** - increase network performance even more by using header and data compression (extensive CPU usage)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

unpacking (*none* | *simple* | *compress-all* | *compress-headers*; default: **simple**) - specifies the unpacking mode

- **none** - accept only usual packets
- **simple** - accept usual packets and aggregated packets without compression
- **compress-headers** - accept all packets except those with payload compression
- **compress-all** - accept all packets

Notes

Level of packet compression increases like this: **none** -> **simple** -> **compress-headers** -> **compress-all**.

When router has to send a packet it choses minimum level of packet compression from what its own **packing** type is set and what other router's **unpacking** type is set. Same is with **aggregated-size** setting - minimum value of both ends is actual maximum size of aggregated packet used.

aggregated-size can be bigger than interface MTU if network device allows it to be (i.e., it supports sending and receiving frames bigger than 1514 bytes)

Example

To enable maximal compression on the **ether1** interface:

```
[admin@MikroTik] ip packing> add interface=ether1 packing=compress-all \
\... unpacking=compress-all
[admin@MikroTik] ip packing> print
Flags: X - disabled
  #   INTERFACE PACKING          UNPACKING        AGGREGATED-SIZE
  0   ether1    compress-all     compress-all     1500

[admin@MikroTik] ip packing>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

MOXA C101 Synchronous Interface

Document revision 1.1 (Fri Mar 05 08:15:42 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
  Additional Documents
Synchronous Interface Configuration
  Description
  Property Description
  Notes
  Example
Troubleshooting
  Description
Synchronous Link Application Examples
  MikroTik Router to MikroTik Router
  MikroTik Router to Cisco Router

General Information

Summary

The MikroTik RouterOS supports MOXA C101 Synchronous 4Mb/s Adapter hardware. The V.35 synchronous interface is the standard for VSAT and other satellite modems. However, you must check with the satellite system supplier for the modem interface type.

Specifications

Packages required: ***synchronous***
License required: ***level4***
Home menu level: ***/interface moxa-c101***
Standards and Technologies: ***Cisco/HDLC-X.25 (RFC 1356)***, ***Frame Relay (RFC1490)***, ***PPP (RFC-1661)***, ***PPP (RFC-1662)***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***Log Management***

Description

You can install up to four MOXA C101 synchronous cards in one PC box, if you have so many slots and IRQs available. Assuming you have all necessary packages and licenses installed, in most cases it should to be done nothing at that point (all drivers are loaded automatically). However, if you have a non Plug-and-Play ISA card, the corresponding driver requires to be loaded.

MOXA C101 PCI variant cabling

The MOXA C101 PCI requires different from MOXA C101 ISA cable. It can be made using the following table:

DB25f	Signal	Direction	V.35m
4	**RTS**	**OUT**	**C**
5	**CTS**	**IN**	**D**
6	**DSR**	**IN**	**E**
7	**GND**	**-**	**B**
8	**DCD**	**IN**	**F**
10	**TxDB**	**OUT**	**S**
11	**TxDA**	**OUT**	**P**
12	**RxDB**	**IN**	**T**
13	**RxDA**	**IN**	**R**
14	**TxCB**	**IN**	**AA**
16	**TxCA**	**IN**	**Y**
20	**DTR**	**OUT**	**H**
22	**RxCB**	**IN**	**X**
23	**RxCA**	**IN**	**V**
short 9 and 25 pin			

Additional Documents

For more information about the MOXA C101 synchronous 4Mb/s adapter hardware please see:

- ***http://www.moxa.com/product/sync/C101.htm*** - the product on-line documentation
- ***C101 SuperSync Board User's Manual*** the user's manual in PDF format

Synchronous Interface Configuration

Home menu level: ***/interface moxa-c101***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

Moxa c101 synchronous interface is shown under the interfaces list with the name moxa-c101-N

Property Description

name (*name*; default: **moxa-c101-N**) - interface name

cisco-hdlc-keepalive-interval (*time*; default: **10s**) - keepalive period in seconds

clock-rate (*integer*; default: **64000**) - speed of internal clock

clock-source (*external* | *internal* | *tx-from-rx* | *tx-internal*; default: **external**) - clock source

frame-relay-dce (*yes* | *no*; default: **no**) - operate or not in DCE mode

frame-relay-lmi-type (*ansi* | *ccitt*; default: **ansi**) - Frame-relay Local Management Interface type:

- **ansi** - set LMI type to ANSI-617d (also known as Annex A)
- **ccitt** - set LMI type to CCITT Q933a (also known as Annex A)

ignore-dcd (*yes* | *no*; default: **no**) - ignore or not DCD

line-protocol (*cisco-hdlc* | *frame-relay* | *sync-ppp*; default: **sync-ppp**) - line protocol name

mtu (*integer*; default: **1500**) - Maximum Transmit Unit

Notes

If you purchased the MOXA C101 Synchronous card from MikroTik, you have received a V.35 cable with it. This cable should work for all standard modems, which have V.35 connections. For synchronous modems, which have a DB-25 connection, you should use a standard DB-25 cable.

The MikroTik driver for the MOXA C101 Synchronous adapter allows you to unplug the V.35 cable from one modem and plug it into another modem with a different clock speed, and you do not need to restart the interface or router.

Example

```
[admin@MikroTik] interface> moxa-c101
[admin@MikroTik] interface moxa-c101> print
Flags: X - disabled, R - running
  0  R name="moxa-c101-1" mtu=1500 line-protocol=sync-ppp clock-rate=64000
       clock-source=external frame-relay-lmi-type=ansi frame-relay-dce=no
       cisco-hdlc-keepalive-interval=10s ignore-dcd=no

[admin@MikroTik] interface moxa-c101>
```

You can monitor the status of the synchronous interface:

```
[admin@MikroTik] interface moxa-c101> monitor 0
    dtr: yes
    rts: yes
    cts: no
    dsr: no
    dcd: no

[admin@MikroTik] interface moxa-c101>
```

Connect a communication device, e.g., a baseband modem, to the V.35 port and turn it on. If the link is working properly the status of the interface is:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] interface moxa-c101> monitor 0
    dtr: yes
    rts: yes
    cts: yes
    dsr: yes
    dcd: yes

[admin@MikroTik] interface moxa-c101>
```

Troubleshooting

Description

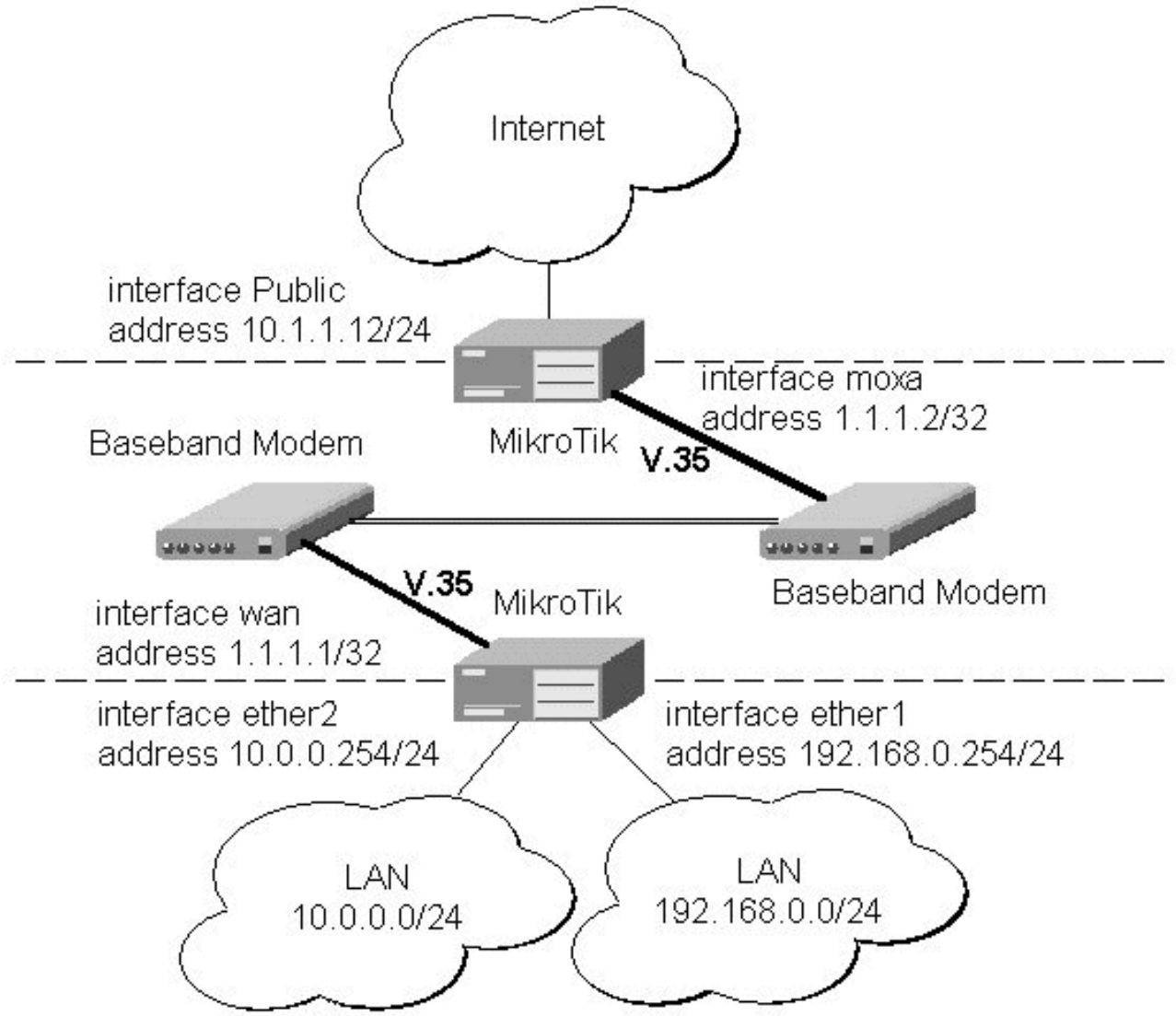
- **The synchronous interface does not show up under the interfaces list**
  Obtain the required license for synchronous feature
- **The synchronous link does not work**
  Check the V.35 cabling and the line between the modems. Read the modem manual

Synchronous Link Application Examples

MikroTik Router to MikroTik Router

Let us consider the following network setup with two MikroTik Routers connected to a leased line with baseband modems:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



The driver for MOXA C101 card should be loaded and the interface should be enabled according to the instructions given above. The IP addresses assigned to the synchronous interface should be as follows:

```
[admin@MikroTik] ip address> add address 1.1.1.1/32 interface wan \
\... network 1.1.1.2 broadcast 255.255.255.255

[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.254/24      10.0.0.254      10.0.0.255      ether2
  1   192.168.0.254/24   192.168.0.254   192.168.0.255   ether1
  2   1.1.1.1/32         1.1.1.2         255.255.255.255 wan
[admin@MikroTik] ip address> /ping 1.1.1.2
1.1.1.2 64 byte pong: ttl=255 time=31 ms
1.1.1.2 64 byte pong: ttl=255 time=26 ms
1.1.1.2 64 byte pong: ttl=255 time=26 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 26/27.6/31 ms
[admin@MikroTik] ip address>
```

The default route should be set to the gateway router 1.1.1.2:

```
[admin@MikroTik] ip route> add gateway 1.1.1.2
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0  S 0.0.0.0/0          r 1.1.1.2         1        wan
    1 DC 10.0.0.0/24        r 10.0.0.254      1        ether2
    2 DC 192.168.0.0/24     r 192.168.0.254   0        ether1
    3 DC 1.1.1.2/32         r 0.0.0.0         0        wan

[admin@MikroTik] ip route>
```

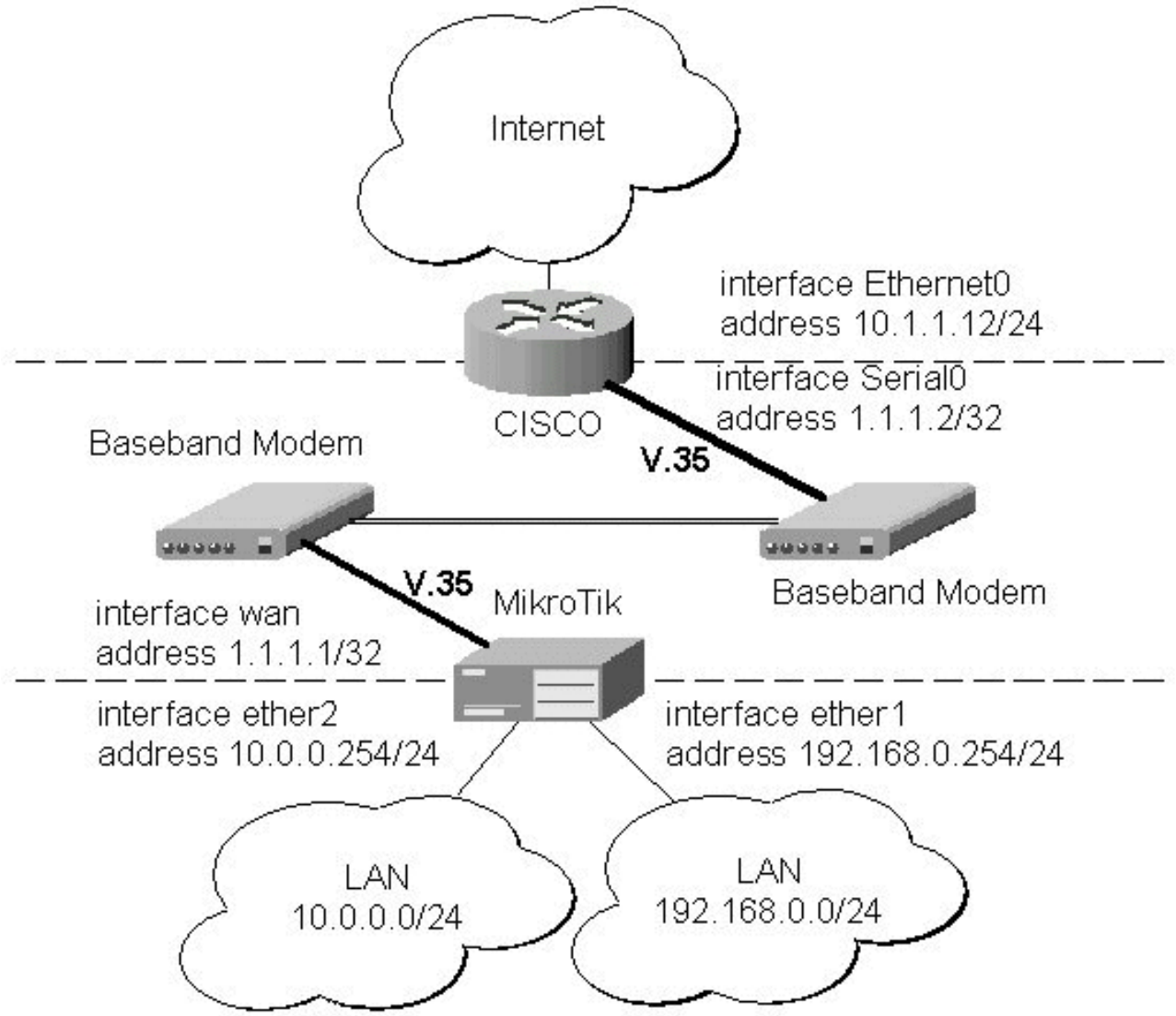
The configuration of the MikroTik router at the other end is similar:

```
[admin@MikroTik] ip address> add address 1.1.1.2/32 interface moxa \
\... network 1.1.1.1 broadcast 255.255.255.255
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.1.1.12/24       10.1.1.12       10.1.1.255      Public
  1   1.1.1.2/32         1.1.1.1         255.255.255.255 moxa
[admin@MikroTik] ip address> /ping 1.1.1.1
1.1.1.1 64 byte pong: ttl=255 time=31 ms
1.1.1.1 64 byte pong: ttl=255 time=26 ms
1.1.1.1 64 byte pong: ttl=255 time=26 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 26/27.6/31 ms
[admin@MikroTik] ip address>
```

MikroTik Router to Cisco Router

Let us consider the following network setup with MikroTik Router connected to a leased line with baseband modems and a CISCO router at the other end:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



The driver for MOXA C101 card should be loaded and the interface should be enabled according to the instructions given above. The IP addresses assigned to the synchronous interface should be as follows:

```
[admin@MikroTik] ip address> add address 1.1.1.1/32 interface wan \
\... network 1.1.1.2 broadcast 255.255.255.255
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.254/24      10.0.0.254      10.0.0.255      ether2
  1   192.168.0.254/24   192.168.0.254   192.168.0.255   ether1
  2   1.1.1.1/32         1.1.1.2         255.255.255.255 wan
[admin@MikroTik] ip address> /ping 1.1.1.2
1.1.1.2 64 byte pong: ttl=255 time=31 ms
1.1.1.2 64 byte pong: ttl=255 time=26 ms
1.1.1.2 64 byte pong: ttl=255 time=26 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 26/27.6/31 ms
[admin@MikroTik] ip address>
```

The default route should be set to the gateway router 1.1.1.2:

```
[admin@MikroTik] ip route> add gateway 1.1.1.2
[admin@MikroTik] ip route> print
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0  S 0.0.0.0/0          r 1.1.1.2         1        wan
    1 DC 10.0.0.0/24        r 10.0.0.254      0        ether2
    2 DC 192.168.0.0/24     r 192.168.0.254   0        ether1
    3 DC 1.1.1.2/32         r 1.1.1.1         0        wan

[admin@MikroTik] ip route>
```

The configuration of the Cisco router at the other end (part of the configuration) is:

```
CISCO#show running-config
Building configuration...

Current configuration:
...
!
interface Ethernet0
 description connected to EthernetLAN
 ip address 10.1.1.12 255.255.255.0
!
interface Serial0
 description connected to MikroTik
 ip address 1.1.1.2 255.255.255.252
 serial restart-delay 1
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.1.1.254
!
...
end

CISCO#
```

Send ping packets to the MikroTik router:

```
CISCO#ping 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/32/40 ms
CISCO#
```

Note! Keep in mind that for the point-to-point link the network mask is set to **32** bits, the argument **network** is set to the IP address of the other end, and the broadcast address is set to **255.255.255.255**.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

MOXA C502 Dual-port Synchronous Interface

Document revision 1.1 (Fri Mar 05 08:16:21 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
  Additional Documents
Synchronous Interface Configuration
  Description
  Property Description
  Notes
  Example
Troubleshooting
  Description
Synchronous Link Application Examples
  MikroTik Router to MikroTik Router
  MikroTik Router to Cisco Router

General Information

Summary

The MikroTik RouterOS supports the MOXA C502 PCI Dual-port Synchronous 8Mb/s Adapter hardware. The V.35 synchronous interface is the standard for VSAT and other satellite modems. However, you must check with the satellite system supplier for the modem interface type.

Specifications

Packages required: ***synchronous***
License required: ***level4***
Home menu level: ***/interface moxa-c502***
Standards and Technologies: ***Cisco/HDLC-X.25 (RFC 1356)***, ***Frame Relay (RFC1490)***, ***PPP (RFC-1661)***, ***PPP (RFC-1662)***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***Log Management***

Description

You can install up to four MOXA C502 synchronous cards in one PC box, if you have so many PCI slots available. Assuming you have all necessary packages and licences installed, in most cases it should to be done nothing at that point (all drivers are loaded automatically).

Additional Documents

For more information about the MOXA C502 Dual-port Synchronous 8Mb/s Adapter hardware please see:

- ***http://www.moxa.com/product/sync/C502.htm*** - the product on-line documentation
- ***C502 Dual Port Sync Board User's Manuall*** the user's manual in PDF format

Synchronous Interface Configuration

Home menu level: ***/interface moxa-c502***

Description

Moxa c502 synchronous interface is shown under the interfaces list with the name moxa-c502-N

Property Description

name (*name*; default: **moxa-c502-N**) - interface name

cisco-hdlc-keepalive-interval (*time*; default: **10s**) - keepalive period in seconds

clock-rate (*integer*; default: **64000**) - speed of internal clock

clock-source (*external* | *internal* | *tx-from-rx* | *tx-internal*; default: **external**) - clock source

frame-relay-dce (*yes* | *no*; default: **no**) - operate or not in DCE mode

frame-relay-lmi-type (*ansi* | *ccitt*; default: **ansi**) - Frame-relay Local Management Interface type:

- **ansi** - set LMI type to ANSI-617d (also known as Annex A)
- **ccitt** - set LMI type to CCITT Q933a (also known as Annex A)

ignore-dcd (*yes* | *no*; default: **no**) - ignore or not DCD

line-protocol (*cisco-hdlc* | *frame-relay* | *sync-ppp*; default: **sync-ppp**) - line protocol name

mtu (*integer*; default: **1500**) - Maximum Transmit Unit

Notes

There will be TWO interfaces for each MOXA C502 card since the card has TWO ports.

The MikroTik driver for the MOXA C502 Dual Synchronous adapter allows you to unplug the V.35 cable from one modem and plug it into another modem with a different clock speed, and you do not need to restart the interface or router.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

```
[admin@MikroTik] interface> moxa-c502
[admin@MikroTik] interface moxa-c502> print
Flags: X - disabled, R - running
  0  R name="moxa-c502-1" mtu=1500 line-protocol=sync-ppp clock-rate=64000
       clock-source=external frame-relay-lmi-type=ansi frame-relay-dce=no
       cisco-hdlc-keepalive-interval=10s
  1  R name="moxa-c502-2" mtu=1500 line-protocol=sync-ppp clock-rate=64000
       clock-source=external frame-relay-lmi-type=ansi frame-relay-dce=no
       cisco-hdlc-keepalive-interval=10s

[admin@MikroTik] interface moxa-c502>
```

You can monitor the status of the synchronous interface:

```
[admin@MikroTik] interface moxa-c502> monitor 0
    dtr: yes
    rts: yes
    cts: no
    dsr: no
    dcd: no

[admin@MikroTik] interface moxa-c502>
```

Connect a communication device, e.g., a baseband modem, to the V.35 port and turn it on. If the link is working properly the status of the interface is:

```
[admin@MikroTik] interface moxa-c502> monitor 0
    dtr: yes
    rts: yes
    cts: yes
    dsr: yes
    dcd: yes

[admin@MikroTik] interface moxa-c502>
```

Troubleshooting

Description

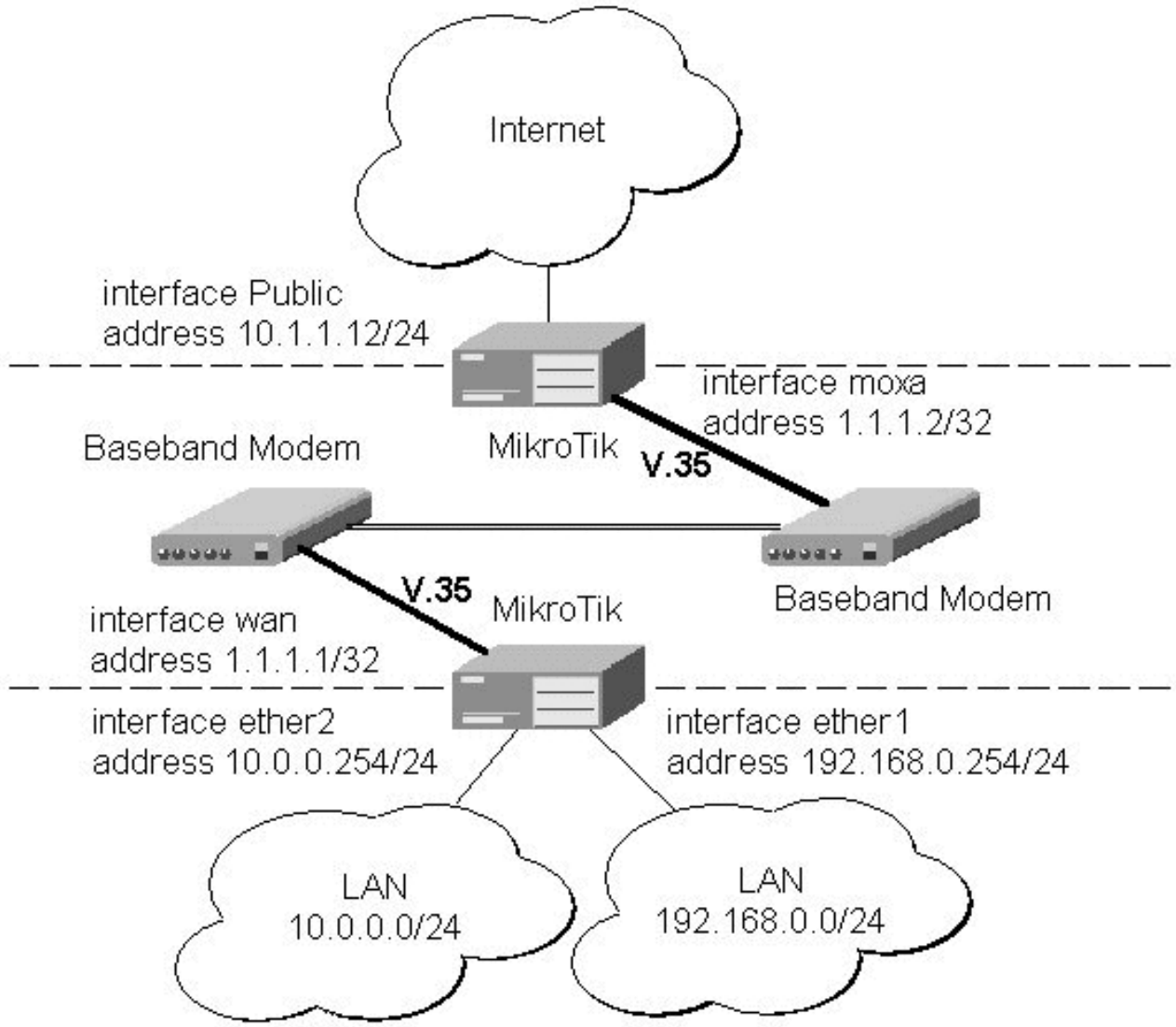
- **The synchronous interface does not show up under the interfaces list**
  Obtain the required license for synchronous feature
- **The synchronous link does not work**
  Check the V.35 cabling and the line between the modems. Read the modem manual

Synchronous Link Application Examples

MikroTik Router to MikroTik Router

Let us consider the following network setup with two MikroTik Routers connected to a leased line with baseband modems:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



The driver for MOXA C502 card should be loaded and the interface should be enabled according to the instructions given above. The IP addresses assigned to the synchronous interface should be as follows:

```
[admin@MikroTik] ip address> add address 1.1.1.1/32 interface wan \
\... network 1.1.1.2 broadcast 255.255.255.255

[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.254/24      10.0.0.254      10.0.0.255      ether2
  1   192.168.0.254/24   192.168.0.254   192.168.0.255   ether1
  2   1.1.1.1/32         1.1.1.2         255.255.255.255 wan
[admin@MikroTik] ip address> /ping 1.1.1.2
1.1.1.2 64 byte pong: ttl=255 time=31 ms
1.1.1.2 64 byte pong: ttl=255 time=26 ms
1.1.1.2 64 byte pong: ttl=255 time=26 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 26/27.6/31 ms
[admin@MikroTik] ip address>
```

The default route should be set to the gateway router 1.1.1.2:

```
[admin@MikroTik] ip route> add gateway 1.1.1.2 interface wan
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0  S 0.0.0.0/0          r 1.1.1.2         1        wan
    1 DC 10.0.0.0/24        r 10.0.0.254      1        ether2
    2 DC 192.168.0.0/24     r 192.168.0.254   0        ether1
    3 DC 1.1.1.2/32         r 0.0.0.0         0        wan

[admin@MikroTik] ip route>
```

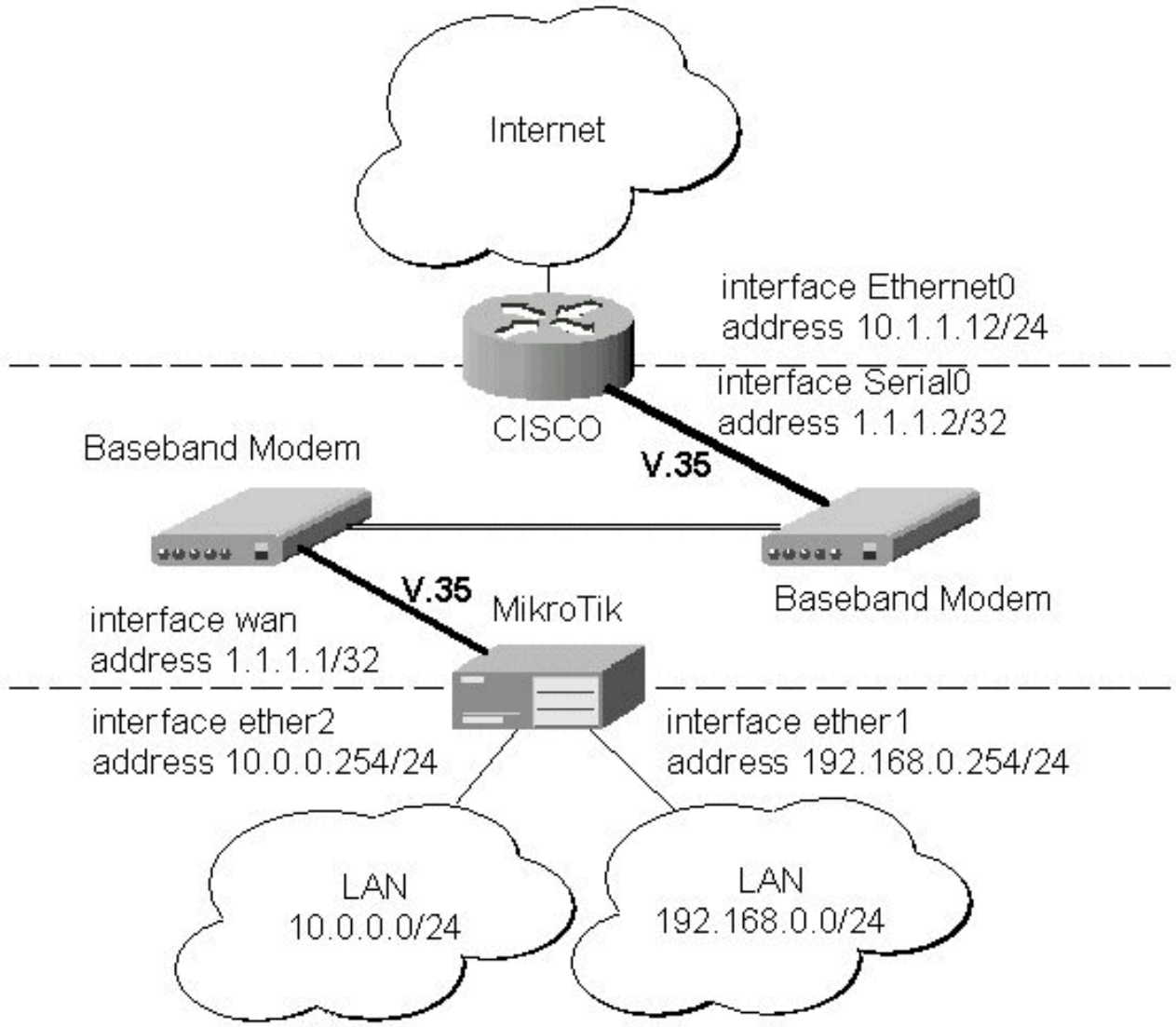
The configuration of the MikroTik router at the other end is similar:

```
[admin@MikroTik] ip address> add address 1.1.1.2/32 interface moxa \
\... network 1.1.1.1 broadcast 255.255.255.255
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.1.1.12/24       10.1.1.12       10.1.1.255      Public
  1   1.1.1.2/32         1.1.1.1         255.255.255.255 moxa
[admin@MikroTik] ip address> /ping 1.1.1.1
1.1.1.1 64 byte pong: ttl=255 time=31 ms
1.1.1.1 64 byte pong: ttl=255 time=26 ms
1.1.1.1 64 byte pong: ttl=255 time=26 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 26/27.6/31 ms
[admin@MikroTik] ip address>
```

MikroTik Router to Cisco Router

Let us consider the following network setup with MikroTik Router connected to a leased line with baseband modems and a CISCO router at the other end:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



The driver for MOXA C502 card should be loaded and the interface should be enabled according to the instructions given above. The IP addresses assigned to the synchronous interface should be as follows:

```
[admin@MikroTik] ip address> add address 1.1.1.1/32 interface wan \
\... network 1.1.1.2 broadcast 255.255.255.255
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.254/24      10.0.0.254      10.0.0.255      ether2
  1   192.168.0.254/24   192.168.0.254   192.168.0.255   ether1
  2   1.1.1.1/32         1.1.1.2         255.255.255.255 wan
[admin@MikroTik] ip address> /ping 1.1.1.2
1.1.1.2 64 byte pong: ttl=255 time=31 ms
1.1.1.2 64 byte pong: ttl=255 time=26 ms
1.1.1.2 64 byte pong: ttl=255 time=26 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 26/27.6/31 ms
[admin@MikroTik] ip address>
```

The default route should be set to the gateway router 1.1.1.2:

```
[admin@MikroTik] ip route> add gateway 1.1.1.2
[admin@MikroTik] ip route> print
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0  S 0.0.0.0/0          r 1.1.1.2         1        wan
    1 DC 10.0.0.0/24        r 10.0.0.254      0        ether2
    2 DC 192.168.0.0/24     r 192.168.0.254   0        ether1
    3 DC 1.1.1.2/32         r 1.1.1.1         0        wan

[admin@MikroTik] ip route>
```

The configuration of the Cisco router at the other end (part of the configuration) is:

```
CISCO#show running-config
Building configuration...

Current configuration:
...
!
interface Ethernet0
 description connected to EthernetLAN
 ip address 10.1.1.12 255.255.255.0
!
interface Serial0
 description connected to MikroTik
 ip address 1.1.1.2 255.255.255.252
 serial restart-delay 1
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.1.1.254
!
...
end

CISCO#
```

Send ping packets to the MikroTik router:

```
CISCO#ping 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/32/40 ms
CISCO#
```

Note! Keep in mind that for the point-to-point link the network mask is set to **32** bits, the argument **network** is set to the IP address of the other end, and the broadcast address is set to **255.255.255.255**.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

PPP and Asynchronous Interfaces

Document revision 1.1 (Fri Mar 05 08:16:45 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Additional Documents
Serial Port Configuration
  Property Description
  Notes
  Example
PPP Server Setup
  Description
  Property Description
  Example
PPP Client Setup
  Description
  Property Description
  Notes
  Example
PPP Application Example
  Client - Server Setup

General Information

Summary

PPP (Point-to-Point Protocol) provides a method for transmitting datagrams over serial point-to-point links. Physically it relies on **com1** and **com2** ports from standard PC hardware configurations. These appear as **serial0** and **serial1** automatically. You can add more serial ports to use the router for a modem pool using these adapters:

- MOXA (***http://www.moxa.com***) Smartio CP-132 2-port PCI multiport asynchronous board with maximum of 8 ports (4 cards)
- MOXA (***http://www.moxa.com***) Smartio C104H, CP-114 or CT-114 4-port PCI multiport asynchronous board with maximum of 16 ports (4 cards)
- MOXA (***http://www.moxa.com***) Smartio C168H, CP-168H or CP-168U 8-port PCI multiport asynchronous board with maximum of 32 ports (4 cards)
- Cyclades (***http://www.cyclades.com***) Cyclom-Y Series 4 to 32 port PCI multiport asynchronous board with maximum of 128 ports (4 cards)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- Cyclades (***http://www.cyclades.com***) Cyclades-Z Series 16 to 64 port PCI multiport asynchronous board with maximum of 256 ports (4 cards)
- TCL (***http://www.thetcl.com***) DataBooster 4 or 8 port High Speed Buffered PCI Communication Controllers

Specifications

Packages required: ***ppp***
License required: ***level1***
Home menu level: ***/interface ppp-client, /interface ppp-server***
Standards and Technologies: ***PPP (RFC 1661)***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***
- ***Log Management***
- ***AAA***

Additional Documents

- ***http://www.ietf.org/rfc/rfc2138.txt?number=2138***
- ***http://www.ietf.org/rfc/rfc2138.txt?number=2139***

Serial Port Configuration

Home menu level: ***/port***

Property Description

name (*name*; default: **serialN**) - port name
used-by (*read-only: text*) - shows the user of the port. Only free ports can be used in PPP setup
baud-rate (*integer*; default: **9600**) - maximal data rate of the port
data-bits (*7* | *8*; default: **8**) - number of bits per character transmitted
parity (*none* | *even* | *odd*; default: **none**) - character parity check method
stop-bits (*1* | *2*; default: **1**) - number of stop bits after each character transmitted
flow-control (*none* | *hardware* | *xon-xoff*; default: **hardware**) - flow control method

Notes

Keep in mind that **baud-rate**, **data-bits**, **parity**, **stop-bits** and **flow control** parameters must be the same for both communicating sides.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

```
[admin@MikroTik] > /port print
  # NAME                              USED-BY                           BAUD-RATE
  0 serial0                           Serial Console                    9600
  1 databooster1                                                        9600
  2 databooster2                                                        9600
  3 databooster3                                                        9600
  4 databooster4                                                        9600
  5 databooster5                                                        9600
  6 databooster6                                                        9600
  7 databooster7                                                        9600
  8 databooster8                                                        9600
  9 cycladesA1                                                          9600
 10 cycladesA2                                                          9600
 11 cycladesA3                                                          9600
 12 cycladesA4                                                          9600
 13 cycladesA5                                                          9600
 14 cycladesA6                                                          9600
 15 cycladesA7                                                          9600
 16 cycladesA8                                                          9600
[admin@MikroTik] > set 9 baud-rate=38400
[admin@MikroTik] >
```

PPP Server Setup

Home menu level: ***/interface ppp-server***

Description

PPP server provides a remode connection service for users. When dialing in, the users can be authenticated locally using the local user database in the **/user** menu, or at the RADIUS server specified in the **/ip ppp** settings.

Property Description

port (*name*; default: **(unknown)**) - serial port

authentication (*multiple choice: mschap2, mschap1, chap, pap*; default: **mschap2, mschap1, chap, pap**) - authentication protocol

profile (*name*; default: **default**) - profile name used for the link

mtu (*integer*; default: **1500**) - Maximum Transmission Unit. Maximum packet size to be transmitted

mru (*integer*; default: **1500**) - Maximum Receive Unit

null-modem (*no* | *yes*; default: **no**) - enable/disable null-modem mode (when enabled, no modem initialization strings are sent)

modem-init (*text*; default: **""**) - modem initialization string. You may use "s11=40" to improve dialing speed

ring-count (*integer*; default: **1**) - number of rings to wait before answering phone

name (*name*; default: **ppp-inN**) - interface name for reference

Example

You can add a PPP server using the **add** command:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] interface ppp-server> add name=test port=serial1
[admin@MikroTik] interface ppp-server> print
Flags: X - disabled, R - running
  0 X  name="test" mtu=1500 mru=1500 port=serial1
       authentication=mschap2,chap,pap profile=default modem-init=""
       ring-count=1 null-modem=no

[admin@MikroTik] interface ppp-server> enable 0
[admin@MikroTik] interface ppp-server> monitor test
           status: "waiting for call..."

[admin@MikroTik] interface ppp-server>
```

PPP Client Setup

Home menu level: ***/interface ppp-client***

Description

The section describes PPP clients configuration routines.

Property Description

port (*name*; default: **(unknown)**) - serial port

user (*text*; default: **""**) - P2P user name on the remote server to use for dialout

password (*text*; default: **""**) - P2P user password on the remote server to use for dialout

profile (*name*; default: **default**) - local profile to use for dialout

allow (*multiple choice: mschap2, mschap1, chap, pap*; default: **mschap2, mschap1, chap, pap**) - the protocol to allow the client to use for authentication

phone (*integer*; default: **""**) - phone number for dialout

tone-dial (*yes* | *no*; default: **yes**) - defines whether use tone dial or pulse dial

mtu (*integer*; default: **1500**) - Maximum Transmission Unit. Maximum packet size to be transmitted

mru (*integer*; default: **1500**) - Maximum Receive Unit

null-modem (*no* | *yes*; default: **no**) - enable/disable null-modem mode (when enabled, no modem initialization strings are sent)

modem-init (*text*; default: **""**) - modem initialization strings. You may use "s11=40" to improve dialing speed

dial-on-demand (*yes* | *no*; default: **no**) - enable/disable dial on demand

add-default-route (*yes* | *no*; default: **no**) - add PPP remote address as a default route

use-peer-dns (*yes* | *no*; default: **no**) - use DNS server settings from the remote server

Notes

- Additional client profiles must be configured on the server side for clients to accomplish logon procedure. For more information see **Related Documents** section.
- PPP client profiles must match at least partially (**local-address** and values related to encryption should match) with corresponding remote server values.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

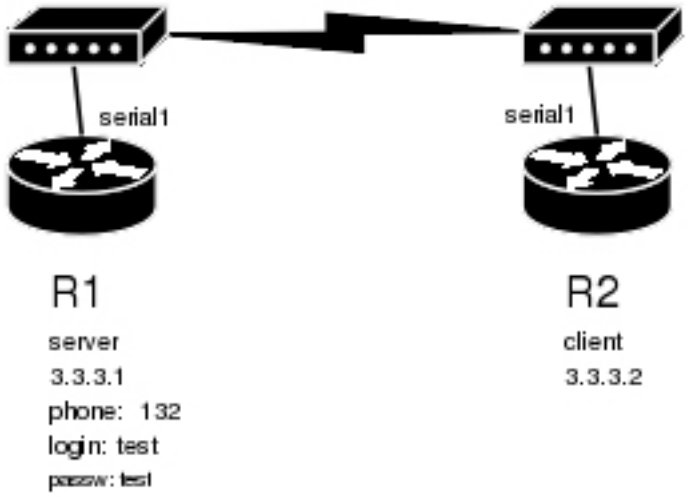
You can add a PPP client using the **add** command:

```
[admin@MikroTik] interface ppp-client> add name=test user=test port=serial1 \
\... add-default-route=yes
[admin@MikroTik] interface ppp-client> print
Flags: X - disabled, R - running
  0 X  name="test" mtu=1500 mru=1500 port=serial1 user="test" password=""
       profile=default phone="" tone-dial=yes modem-init="" null-modem=no
       dial-on-demand=no add-default-route=yes use-peer-dns=no

[admin@MikroTik] interface ppp-client> enable 0
[admin@MikroTik] interface ppp-client> monitor test
[admin@MikroTik] interface ppp-client> monitor 0
           status: "dialing out..."

[admin@MikroTik] interface ppp-client>
```

PPP Application Example

Client - Server Setup

In this example we will consider the following network setup:



For a typical server setup we need to add one user to the **R1** and configure the PPP server.

```
[admin@MikroTik] ppp secret> add name=test password=test local-address=3.3.3.1 \
\... remote-address=3.3.3.2
[admin@MikroTik] ppp secret> print
Flags: X - disabled
  0   name="test" service=any caller-id="" password="test" profile=default
      local-address=3.3.3.1 remote-address=3.3.3.2 routes=""

[admin@MikroTik] ppp secret> /int ppp-server
[admin@MikroTik] interface ppp-server> add port=serial1 disabled=no
[admin@MikroTik] interface ppp-server> print
Flags: X - disabled, R - running
  0    name="ppp-in1" mtu=1500 mru=1500 port=serial1
       authentication=mschap2,mschap1,chap,pap profile=default modem-init=""
       ring-count=1 null-modem=no

[admin@MikroTik] interface ppp-server>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Now we need to setup the client to connect to the server:

```
[admin@MikroTik] interface ppp-client> add port=serial1 user=test password=test \
\... phone=132
[admin@MikroTik] interface ppp-client> print
Flags: X - disabled, R - running
  0 X  name="ppp-out1" mtu=1500 mru=1500 port=serial1 user="test"
       password="test" profile=default phone="132" tone-dial=yes
       modem-init="" null-modem=no dial-on-demand=no add-default-route=no
       use-peer-dns=no

[admin@MikroTik] interface ppp-client> enable 0

After a short duration of time the routers will be able to ping each other:
[admin@MikroTik] interface ppp-client> /ping 3.3.3.1
3.3.3.1 64 byte ping: ttl=64 time=43 ms
3.3.3.1 64 byte ping: ttl=64 time=11 ms
3.3.3.1 64 byte ping: ttl=64 time=12 ms
3.3.3.1 64 byte ping: ttl=64 time=11 ms
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 11/19.2/43 ms
[admin@MikroTik] interface ppp-client>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

RadioLAN 5.8GHz Wireless Interface

Document revision 1.1 (Fri Mar 05 08:17:04 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
Wireless Interface Configuration
  Description
  Property Description
  Example
Troubleshooting
  Description
Wireless Network Applications
  Point-to-Point Setup with Routing

General Information

Summary

The MikroTik RouterOS supports the following RadioLAN 5.8GHz Wireless Adapter hardware:

- RadioLAN ISA card (Model 101)
- RadioLAN PCMCIA card

For more information about the RadioLAN adapter hardware please see the relevant User???s Guides and Technical Reference Manuals.

Specifications

Packages required: ***radiolan***
License required: ***level4***
Home menu level: ***/interface radiolan***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***
- ***Log Management***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

Installing the Wireless Adapter

These installation instructions apply to non-Plug-and-Play ISA cards. If You have a Plug-and-Play compliant system AND **PnP OS Installed** option in system BIOS is set to **Yes** AND you have a Plug-and-Play compliant ISA or PCI card (using PCMCIA or CardBus card with Plug-and-Play compliant adapter), the driver should be loaded automatically. If it is not, these instructions may also apply to your system.

The basic installation steps of the wireless adapter should be as follows:

1. Check the system BIOS settings for peripheral devices, like, Parallel or Serial communication ports. Disable them, if you plan to use IRQ's assigned to them by the BIOS.
2. Use the ***RLProg.exe*** to set the IRQ and Base Port address of the RadioLAN ISA card (Model 101). RLProg must not be run from a DOS window. Use a separate computer or a bootable floppy to run the RLProg utility and set the hardware parameters. The factory default values of I/O 0x300 and IRQ 10 might conflict with other devices.

Please note, that not all combinations of I/O base addresses and IRQs may work on your motherboard. As it has been observed, the IRQ 5 and I/O 0x300 work in most cases.

Wireless Interface Configuration

Home menu level: ***/interface ratiolan***

Description

To set the wireless interface for working with another wireless card in a point-to-point link, you should set the following parameters:

- The **Service Set Identifier**. It should match the sid of the other card.
- The **Distance** should be set to that of the link. For example, if you have 6 km link, use distance 4.7 km - 6.6 km.

All other parameters can be left as default. You can monitor the list of neighbors having the same sid and being within the radio range.

Property Description

name (*name*; default: **radiolanN**) - assigned interface name

mtu (*integer*; default: **1500**) - Maximum Transmission Unit

mac-address (*read-only: MAC address*) - MAC address

distance (*0-150m* | *10.2km-13.0km* | *2.0km-2.9km* | *4.7km-6.6km* | *1.1km-2.0km* | *150m-1.1km* | *2.9km-4.7km* | *6.6km-10.2km*; default: **0-150m**) - distance setting for the link

rx-diversity (*enabled* | *disabled*; default: **disabled**) - receive diversity

tx-diversity (*enabled* | *disabled*; default: **disabled**) - transmit diversity

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

default-destination (*ap* | *as-specified* | *first-ap* | *first-client* | *no-destination*; default: **first-client**) - default destination. It sets the destination where to send the packet if it is not for a client in the radio network

default-address (*MAC address*; default: **00:00:00:00:00:00**) - MAC address of a host in the radio network where to send the packet, if it is for none of the radio clients

max-retries (*integer*; default: **1500**) - maximum retries before dropping the packet

sid (*text*) - Service Identifier

card-name (*text*) - card name

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol, one of the:

- **disabled** - the interface will not use ARP protocol
- **enabled** - the interface will use ARP protocol
- **proxy-arp** - the interface will be an ARP proxy (see corresponding manual)
- **reply-only** - the interface will only reply to the requests originated to its own IP addresses, but neighbor MAC addresses will be gathered from /ip arp statically set table only.

Example

```
[admin@MikroTik] interface radiolan> print
Flags: X - disabled, R - running
  0  R name="radiolan1" mtu=1500 mac-address=00:A0:D4:20:4B:E7 arp=enabled
       card-name="00A0D4204BE7" sid="bbbb" default-destination=first-client
       default-address=00:00:00:00:00:00 distance=0-150m max-retries=15
       tx-diversity=disabled rx-diversity=disabled


[admin@MikroTik] interface radiolan>
```

You can monitor the status of the wireless interface:

```
[admin@MikroTik] interface radiolan> monitor radiolan1
    default: 00:00:00:00:00:00
      valid: no

[admin@MikroTik] interface radiolan>
```

Here, the wireless interface card has not found any neighbor.

```
[admin@MikroTik] interface radiolan> set 0 sid ba72 distance 4.7km-6.6km
[admin@MikroTik] interface radiolan> print
Flags: X - disabled, R - running
  0  R name="radiolan1" mtu=1500 mac-address=00:A0:D4:20:4B:E7 arp=enabled
       card-name="00A0D4204BE7" sid="ba72" default-destination=first-client
       default-address=00:00:00:00:00:00 distance=4.7km-6.6km max-retries=15
       tx-diversity=disabled rx-diversity=disabled

[admin@MikroTik] interface radiolan> monitor 0
    default: 00:A0:D4:20:3B:7F
      valid: yes

[admin@MikroTik] interface radiolan>
```

Now we'll monitor other cards with the same **sid** within range:

```
[admin@MikroTik] interface radiolan> neighbor radiolan1 print
Flags: A - access-point, R - registered, U - registered-to-us,
D - our-default-destination
      NAME                ADDRESS           ACCESS-POINT
    D 00A0D4203B7F        00:A0:D4:20:3B:7F
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] interface radiolan>
```

You can test the link by pinging the neighbor by its MAC address:

```
[admin@MikroTik] interface radiolan> ping 00:a0:d4:20:3b:7f radiolan1 \
\... size=1500 count=50
                 sent: 1
    successfully-sent: 1
           max-retries: 0
       average-retries: 0
           min-retries: 0

                 sent: 11
    successfully-sent: 11
           max-retries: 0
       average-retries: 0
           min-retries: 0

                 sent: 21
    successfully-sent: 21
           max-retries: 0
       average-retries: 0
           min-retries: 0

                 sent: 31
    successfully-sent: 31
           max-retries: 0
       average-retries: 0
           min-retries: 0

                 sent: 41
    successfully-sent: 41
           max-retries: 0
       average-retries: 0
           min-retries: 0

                 sent: 50
    successfully-sent: 50
           max-retries: 0
       average-retries: 0
           min-retries: 0

[admin@MikroTik] interface radiolan>
```

Troubleshooting

Description

- **The radiolan interface does not show up under the interfaces list**
  Obtain the required license for RadioLAN 5.8GHz wireless feature
- **The wireless card does not obtain the MAC address of the default destination**
  Check the cabling and antenna alignment

Wireless Network Applications

Point-to-Point Setup with Routing

Let us consider the following network setup:

The minimum configuration required for the RadioLAN interfaces of both routers is:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

1. Setting the Service Set Identifier (up to alphanumeric characters). In our case we use SSID "ba72"
2. Setting the distance parameter, in our case we have 6km link.

The IP addresses assigned to the wireless interface of Router#1 should be from the network 10.1.0.0/30, e.g.:

```
[admin@MikroTik] ip address> add address=10.1.0.1/30 interface=radiolan1
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.1.1.12/24       10.1.1.0        10.1.1.255      ether1
  1   10.1.0.1/30        10.1.0.0        10.1.0.3        radiolan1
[admin@MikroTik] ip address>
```

The default route should be set to the gateway router 10.1.1.254. A static route should be added for the network 192.168.0.0/24:

```
[admin@MikroTik] ip route> add gateway=10.1.1.254
comment  copy-from  disabled  distance  dst-address  netmask  preferred-source
[admin@MikroTik] ip route> add gateway=10.1.1.254 preferred-source=10.1.0.1
[admin@MikroTik] ip route> add dst-address=192.168.0.0/24 gateway=10.1.0.2 \
\... preferred-source=10.1.0.1
[admin@MikroTik] ip route> print
Flags: X - disabled, I - invalid, D - dynamic, J - rejected,
C - connect, S - static, R - rip, O - ospf, B - bgp
    #    DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
    0  S 0.0.0.0/0          u 10.1.1.254      1        radiolan1
    1  S 192.168.0.0/24     r 10.1.0.2        1        radiolan1
    2 DC 10.1.0.0/30        r 0.0.0.0         0        radiolan1
    3 DC 10.1.1.0/24        r 0.0.0.0         0        ether1
[admin@MikroTik] ip route>
```

The Router#2 should have addresses 10.1.0.2/30 and 192.168.0.254/24 assigned to the radiolan and Ethernet interfaces respectively. The default route should be set to 10.1.0.1

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Wireless Client and Wireless Access Point Manual

Document revision 2 (Mon Aug 08 07:47:54 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Quick Setup Guide
  Specifications
  Related Documents
  Description
Wireless Interface Configuration
  Description
  Property Description
  Notes
  Example
Nstreme Settings
  Description
  Property Description
  Notes
  Example
Nstreme2 Group Settings
  Description
  Property Description
  Notes
  Example
Registration Table
  Description
  Property Description
  Example
Connect List
  Description
  Property Description
Access List
  Description
  Property Description
  Notes
  Example
Info
  Description
  Property Description
  Notes
  Example
Virtual Access Point Interface
  Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Property Description
WDS Interface Configuration
Description
Property Description
Notes
Example
Align
Description
Property Description
Notes
Example
Align Monitor
Description
Property Description
Example
Frequency Monitor
Description
Property Description
Example
Manual Transmit Power Table
Description
Property Description
Example
Network Scan
Description
Property Description
Example
Security Profiles
Description
Property Description
Notes
Sniffer
Description
Property Description
Sniffer Sniff
Description
Property Description
Command Description
Sniffer Packets
Description
Property Description
Example
Snooper
Description
Property Description
Command Description
Example
Station and AccessPoint
WDS Station

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Virtual Access Point
Nstreme
Dual Nstreme
WEP Security
Troubleshooting
Description

General Information

Summary

The wireless interface operates using IEEE 802.11 set of standards. It uses radio waves as a physical signal carrier and is capable of wireless data transmission with speeds up to 108 Mbps (in 5GHz turbo-mode).

MikroTik RouterOS supports the Intersil Prism II PC/PCI, Atheros AR5000, AR5001X, AR5001X+, AR5002X+, AR5004X+ and AR5006 chipset based wireless adapter cards for working as wireless clients (**station** mode), wireless bridges (**bridge** mode), wireless access points (**ap-bridge** mode), and for antenna positioning (**alignment-only** mode). For furher information about supported wireless adapters, see ***Device Driver List***

MikroTik RouterOS provides a complete support for IEEE 802.11a, 802.11b and 802.11g wireless networking standards. There are several features implemented for the wireless data communication in RouterOS - WPA (Wi-Fi Protected Access), WEP (Wired Equivalent Privacy), WDS (Wireless Distribution System), DFS (Dynamic Frequency Selection), Alignment mode (for positioning antennas and monitoring wireless signal), disable packet forwarding among clients, and others.

The nstreme protocol is MikroTik proprietary (i.e., incompatible with other vendors) wireless protocol created to improve point-to-point and point-to-multipoint wireless links. Nstreme2 works with a pair of wireless cards (Atheros AR5210 and newer MAC chips only) - one for transmitting data and one for receiving.

Benefits of nstreme protocol:

- Client polling
- Very low protocol overhead per frame allowing super-high data rates
- No protocol limits on link distance
- No protocol speed degradation for long link distances
- Dynamic protocol adjustment depending on traffic type and resource usage

Quick Setup Guide

Let's consider that you have a wireless interface, called **wlan1**.

- To set it as an Access Point, working in 802.11g standard, using frequency **2442 MHz** and Service Set Identifier **test**, do the following configuration:

```
/interface wireless set wlan1 ssid=test frequency=2442 band=2.4ghz-b/g \
   mode=ap-bridge disabled=no
```

  Now your router is ready to accept wireless clients.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- To make a point-to-point connection, using 802.11a standard, frequency **5805 MHz** and Service Set Identifier **p2p**, write:

```
/interface wireless set wlan1 ssid="p2p" frequency=5805 band=5ghz \
    mode=bridge disabled=no
```

  The remote interface should be configured to station as showed below.

- To make the wireless interface as a wireless station, working in 802.11a standard and Service Set Identifier **p2p**:

```
/interface wireless set wlan1 ssid="p2p" band=5ghz mode=station disabled=no
```

Specifications

Packages required: ***wireless***
License required: ***level4 (station and bridge mode), level5 (station, bridge and AP mode), levelfreq (more frequencies)***
Home menu level: ***/interface wireless***
Standards and Technologies: ***IEEE802.11a***, ***IEEE802.11b***, ***IEEE802.11g***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***
-

Description

The Atheros card has been tested for distances up to 20 km providing connection speed up to 17Mbit/s. With appropriate antennas and cabling the maximum distance should be as far as 50 km.

These values of **ack-timeout** were approximated from the tests done by us, as well as by some of our customers:

range	ack-timeout		
	5GHz	5GHz-turbo	2.4GHz-G
0km	default	default	default
5km	52	30	62
10km	85	48	96
15km	121	67	133
20km	160	89	174
25km	203	111	219
30km	249	137	368
35km	298	168	320

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

40km	**350**	**190**	**375**
45km	**405**	**-**	**-**

Please **note** that these are not the precise values. Depending on hardware used and many other factors they may vary up to +/- 15 microseconds.

You can also use **dynamic** ack-timeout value - the router will determine **ack-timeout** setting automatically by sending periodically packets with a different ack-timeout. Ack-timeout values by which ACK frame was received are saved and used later to determine the real ack-timeout.

The Nstreme protocol may be operated in three modes:

- **Point-to-Point mode** - controlled point-to-point mode with one radio on each side
- **Dual radio Point-to-Point mode (Nstreme2)** - the protocol will use two radios on both sides simultaneously (one for transmitting data and one for receiving), allowing superfast point-to-point connection
- **Point-to-Multipoint** - controlled point-to-multipoint mode with client polling (like AP-controlled TokenRing)

Hardware Notes

The MikroTik RouterOS supports as many Atheros chipset based cards as many free adapter slots are on your system. One license is valid for all cards on your system. **Note** that maximal number of PCMCIA sockets is 8.

Some chipsets are not stable with Atheros cards and cause radio to stop working. MikroTik RouterBoard 200, RouterBoard 500 series, and systems based on Intel i815 and i845 chipsets are tested and work stable with Atheros cards. There might be many other chipsets that are working stable, but it has been reported that some older chipsets, and some systems based on AMD Duron CPU are not stable.

Only AR5212 and newer Atheros MAC chips are stable with RouterBOARD200 connected via RouterBOARD14 four-port MiniPCI-to-PCI adapter. This note applies only to the RouterBOARD200 platform with Atheros-based cards.

Wireless Interface Configuration

Home menu level: ***/interface wireless***

Description

In this section we will discuss the most important part of the configuration.

Property Description

ack-timeout (*integer* | *dynamic* | *indoors*) - acknowledgement code timeout (transmission acceptance timeout) in microseconds for acknowledgement messages. Can be one of these:

- **dynamic** - ack-timeout is chosen automatically
- **indoors** - standard constant for indoor usage

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

antenna-gain (*integer*; default: **0**) - antenna gain in dBi. This parameter will be used to calculate whether your system meets regulatory domain's requirements in your country

antenna-mode (*ant-a* | *ant-b* | *rxa-txb* | *txa-rxb*; default: **ant-a**) - which antenna to use for transmit/receive data:

- **ant-a** - use only antenna a
- **ant-b** - use only antenna b
- **rxa-txb** - use antenna a for receiving packets, use antenna b for transmitting packets
- **txa-rxb** - use antenna a for transmitting packets, antenna b for receiving packets

area (*text*; default: **""**) - a string value which is used to describe an access point. It is a value by which a station chooses whether to connect to it or not, using area-prefix parameter from connect-list

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol setting

band - operating band

- **2.4ghz-b** - IEEE 802.11b
- **2.4ghz-b/g** - IEEE 802.11g (supports also IEEE 802.11b)
- **2.4ghz-g-turbo** - IEEE 802.11g up to 108 Mbit
- **2.4ghz-onlyg** - only IEEE 802.11g
- **5ghz** - IEEE 802.11a up to 54 Mbit
- **5ghz-turbo** - IEEE 802.11a up to 108Mbit

basic-rates-a/g (*multiple choice: 6Mbps, 9Mbps, 12Mbps, 18Mbps, 24Mbps, 36Mbps, 48Mbps, 54Mbps*; default: **6Mbps**) - basic rates in 802.11a or 802.11g standard (this should be the minimal speed all the wireless network nodes support). It is recommended to leave this as default

basic-rates-b (*multiple choice: 1Mbps, 2Mbps, 5.5Mbps, 11Mbps*; default: **1Mbps**) - basic rates in 802.11b mode (this should be the minimal speed all the wireless network nodes support). It is recommended to leave this as default

burst-time (*time*; default: **disabled**) - time in microseconds which will be used to send data without stopping. Note that other wireless cards in that network will not be able to transmit data for burst-time microseconds. This setting is available only for AR5000, AR5001X, and AR5001X+ chipset based cards

compression (yes | no; default: **no**) - if enabled on AP (in ap-bridge or bridge mode), it advertizes that it is capable to use hardware data compression. If a client, connected to this AP also supports and is configured to use the hardware data compression, it requests the AP to use compression. This property does not affect clients which do not support compression.

country (*albania* | *algeria* | *argentina* | *armenia* | *australia* | *austria* | *azerbaijan* | *bahrain* | *belarus* | *belgium* | *belize* | *bolvia* | *brazil* | *brunei darussalam* | *bulgaria* | *canada* | *chile* | *china* | *colombia* | *costa rica* | *croatia* | *cyprus* | *czech republic* | *denmark* | *dominican republic* | *ecuador* | *egypt* | *el salvador* | *estonia* | *finland* | *france* | *france_res* | *georgia* | *germany* | *greece* | *guatemala* | *honduras* | *hong kong* | *hungary* | *iceland* | *india* | *indonesia* | *iran* | *ireland* | *israel* | *italy* | *japan* | *japan1* | *japan2* | *japan3* | *japan4* | *japan5* | *jordan* | *kazakhstan* | *korea republic* | *korea republic2* | *kuwait* | *latvia* | *lebanon* | *liechtenstein* | *lithuania* | *luxemburg* | *macau* | *macedonia* | *malaysia* | *mexico* | *monaco* | *morocco* | *netherlands* | *new zealand* | *no_country_set* | *north korea* | *norway* | *oman* | *pakistan* | *panama* | *peru* | *philippines* | *poland* | *portugal* | *puerto rico* | *qatar* | *romania* | *russia* | *saudi arabia* | *singapore* | *slovak republic* | *slovenia* | *south africa* | *spain* | *sweden* | *switzerland* |

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

syria | *taiwan* | *thailand* | *trinidad & tobago* | *tunisia* | *turkey* | *ukraine* | *united arab emirates* | *united kingdom* | *united states* | *uruguay* | *uzbekistan* | *venezuela* | *viet nam* | *yemen* | *zimbabwe*; default: **no_country_set**) - limits wireless settings (frequency and transmit power) to those which are allowed in the respective country

- **no_country_set** - no regulatory domain limitations

default-ap-tx-limit (*integer*; default: **0**) - limits data rate for each wireless client (in bps)

- **0** - no limits

default-authentication (yes | no; default: **yes**) - specifies the default action on the clients side for APs that are not in connect list or on the APs side for clients that are not in access list

- **yes** - enables AP to register a client even if it is not in access list. In turn for client it allows to associate with AP not listed in client's connect list

default-client-tx-limit (*integer*; default: **0**) - limits each client's transmit data rate (in bps). Works only if the client is also a MikroTik Router

- **0** - no limits

default-forwarding (*yes* | *no*; default: **yes**) - to use data forwarding by default or not. If set to 'no', the registered clients will not be able to communicate with each other

dfs-mode (*none* | *radar-detect* | *no-radar-detect*; default: **none**) - used for APs to dynamically select frequency at which this AP will operate

- **none** - do not use DFS
- **no-radar-detect** - AP scans channel list from "scan-list" and chooses the frequency which is with the lowest amount of other networks detected
- **radar-detect** - AP scans channel list from "scan-list" and chooses the frequency which is with the lowest amount of other networks detected, if no radar is detected in this channel for 60 seconds, the AP starts to operate at this channel, if radar is detected, the AP continues searching for the next available channel which is with the lowest amount of other networks detected

disable-running-check (*yes* | *no*; default: **no**) - disable running check. If value is set to 'no', the router determines whether the card is up and running - for AP one or more clients have to be registered to it, for station, it should be connected to an AP. This setting affects the records in the routing table in a way that there will be no route for the card that is not running (the same applies to dynamic routing protocols). If set to 'yes', the interface will always be shown as running

disconnect-timeout (*time*; default: **3s**) - only above this value the client device is considered as disconnected

fast-frames (yes | no; default: **no**) - whether to pack smaller packets into a larger one, which makes larger data rates possible

frequency (*integer*) - operating frequency of the card

frequency-mode (*regulatory-domain* | *manual-tx-power* | *superchannel*; default: **superchannel**) - defines which frequency channels to allow

- **regulatory-domain** - channels in configured country only are allowed, and transmit power is limited to what is allowed in that channel in configured country minus configured antenna-gain. Also note that in this mode card will never be configured to higher power than allowed by the respective regulatory domain
- **manual-tx-power** - channels in configured country only are allowed, but transmit power is taken from tx-power setting
- **superchannel** - only possible with superchannel license. In this mode all hardware supported

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

channels are allowed

hide-ssid (*yes* | *no*; default: **no**) - whether to hide ssid or not in the beacon frames:

- **yes** - ssid is not included in the beacon frames. AP replies only to probe-requests with the given ssid
- **no** - ssid is included in beacon frames. AP replies to probe-requests with the given ssid ant to 'broadcast ssid' (empty ssid)

interface-type (*read-only: text*) - adapter type and model

mac-address (*MAC address*) - Media Access Control (MAC) address of the interface

master-interface (*name*) - physical wireless interface name that will be used by Virtual Access Point (VAP) interface

max-station-count (*integer*: 1..2007; default: **2007**) - maximal number of clients allowed to connect to AP. Real life experiments (from our customers) show that 100 clients can work with one AP, using traffic shaping

mode (*alignment-only* | *ap-bridge* | *bridge* | *nstreme-dual-slave* | *sniffer* | *station* | *station-wds* | *wds-slave*; default: **station**) - operating mode:

- **alignment-only** - this mode is used for positioning antennas (to get the best direction)
- **ap-bridge** - the interface is operating as an Access Point
- **bridge** - the interface is operating as a bridge. This mode acts like ap-bridge with the only difference being it allows only one client
- **nstreme-dual-slave** - the interface is used for nstreme-dual mode
- **sniffer** - promiscuous mode of operation of the wireless card. The card captures wireless frames from all existing transmissions and saves them to a file. Additional configuration resides in the /interface wireless sniffer menu
- **station** - the interface is operating as a client
- **station-wds** - the interface is working as a station, but can communicate with a WDS peer
- **wds-slave** - the interface is working as it would work in ap-bridge mode, but it adapts to its WDS peer's frequency if it is changed

mtu (*integer*: 68..1600; default: **1500**) - Maximum Transmission Unit

name (*name*; default: **wlanN**) - assigned interface name

noise-floor-threshold (*integer* | *default*: -128..127; default: **default**) - value in dBm below which we say that it is rather noise than a normal signal

on-fail-retry-time (*time*; default: **100ms**) - time, after which we repeat to communicate with a wireless device, if a data transmission has failed

periodic-calibration (*default* | *disabled* | *enabled*; default: **default**) - to ensure performance of chipset over temperature and environmental changes, the software performs periodic calibration

preamble-mode (*both* | *long* | *short*; default: **both**) - sets the synchronization field in a wireless packet

- **long** - has a long synchronization field in a wireless packet (128 bits). Is compatible with 802.11 standard
- **short** - has a short synchronization field in a wireless packet (56 bits). Is not compatible with 802.11 standard. With short preamble mode it is possible to get slightly higher data rates
- **both** - supports both - short and long preamble

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

prism-cardtype (*30mW | 100mW | 200mW*) - specify the output of the Prism chipset based card

radio-name (*name*) - descriptive name of the card. Only for MikroTik devices

rate-set (*default | configured*) - which rate set to use:

- **default** - basic and supported-rates settings are not used, instead default values are used.
- **configured** - basic and supported-rates settings are used as configured

scan-list (*multiple choice: integer | default*; default: **default**) - the list of channels to scan

- **default** - represents all frequencies, allowed by the regulatory domain (in the respective country). If no country is set, these frequencies are used - for 2.4GHz mode: 2412, 2417, 2422, 2427, 2432, 2437, 2442, 2447, 2452, 2457, 2462, 2467, 2472; for 2.4GHz-g-turbo mode: 2437; for 5GHz mode: 5180, 5200, 5220, 5240, 5260, 5280, 5300, 5320, 5745, 5765, 5785, 5805; for 5GHz-turbo: 5210, 5250, 5290, 5760, 5800

security-profile (*text*; default: **default**) - which security profile to use. Define security profiles under /interface wireless security-profiles where you can setup WPA or WEP wireless security, for further details, see the Security Profiles section of this manual

ssid (*text*; default: **MikroTik**) - Service Set Identifier. Used to separate wireless networks

supported-rates-a/g (*multiple choice: 6Mbps, 9Mbps, 12Mbps, 18Mbps, 24Mbps, 36Mbps, 48Mbps, 54Mbps*) - rates to be supported in 802.11a or 802.11g standard

supported-rates-b (*multiple choice: 1Mbps, 2Mbps, 5.5Mbps, 11Mbps*) - rates to be supported in 802.11b standard

tx-power (*integer*: -30..30; default: **17**) - manually sets the transmit power of the card (in dBm), if tx-power-mode is set to manual or all-rates-fixed (see tx-power-mode description below)

- **default** - default value of the card

tx-power-mode (*all-rates-fixed | card-rates | default | manual-table*; default: **default**) - choose the transmit power mode for the card:

- **all-rates-fixed** - use one transmit power value for all rates, as configured in tx-power
- **card-rates** - use card default rates
- **default** - use the default tx-power
- **manual-table** - use the transmit powers as defined in /interface wireless manual-tx-power-table

update-stats-interval (*time*) - how often to update statistics in /interface wireless registration-table

wds-default-bridge (*name*; default: **none**) - the default bridge for WDS interface. If you use dynamic WDS then it is very useful in cases when wds connection is reset - the newly created dynamic WDS interface will be put in this bridge

wds-ignore-ssid (yes | no; default: **no**) - if set to 'yes', the AP will create WDS links with any other AP in this frequency. If set to 'no' the ssid values must match on both APs

wds-mode (*disabled | dynamic | static*) - WDS mode:

- **disabled** - WDS interfaces are disabled
- **dynamic** - WDS interfaces are created 'on the fly'
- **static** - WDS interfaces are created manually

Notes

It is strongly suggested to leave basic rates at the lowest setting possible.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Using **compression**, the AP can serve approximately 50 clients with compression enabled!

If **disable-running-check** value is set to **no**, the router determines whether the network interface is up and running - in order to show flag **R** for AP, one or more clients have to be registered to it, for station, it should be connected to an AP. If the interface does not appear as running (**R**), its route in the routing table is shown as **invalid**! If set to **yes**, the interface will always be shown as running.

The **tx-power** default setting is the maximum tx-power that the card can use. If you want to use larger tx-rates, you are able to set them, but **do it at your own risk**! Usually, you can use this parameter to reduce the **tx-power**.

You should set **tx-power** property to an appropriate value as many cards do not have their default setting set to the maximal power it can work on. For the cards MikroTik is selling (5G/ABM), 20dBm (100mW) is the maximal power in 5GHz bands and 18dBm (65mW) is the maximal power in 2.4GHz bands.

For different versions of Atheros chipset there are different value range of **ack-timeout** property:

Chipset version	5ghz		5ghz-turbo		2ghz-b		2ghz-g	
	default	max	default	max	default	max	default	max
5000 (5.2GHz only)	30	204	22	102	N/A	N/A	N/A	N/A
5211 (802.11a/b)	30	409	22	204	109	409	N/A	N/A
5212 (802.11a/b/g)	25	409	22	204	30	409	52	409

If the wireless interfaces are put in **nstreme-dual-slave** mode, all configuration will take place in **/interface wireless nstreme-dual** submenu, described further on in this manual. In that case, configuration made in this submenu will be partially ignored. WDS cannot be used together with the Nstreme-dual.

Example

This example shows how configure a wireless client.

To see current interface settings:

```
[admin@MikroTik] interface wireless> print
Flags: X - disabled, R - running
 0    name="wlan1" mtu=1500 mac-address=00:0B:6B:34:54:FB arp=enabled
      disable-running-check=no interface-type=Atheros AR5213
      radio-name="000B6B3454FB" mode=station ssid="MikroTik"
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=2412 band=2.4ghz-b scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=00:00:03
      on-fail-retry-time=00:00:00.100 preamble-mode=both
[admin@MikroTik] interface wireless>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Set the **ssid** to *mmt*, **band** to *2.4-b/g* and enable the interface. Use the monitor command to see the connection status.

```
[admin@MikroTik] interface wireless> set 0 ssid=mmt disabled=no \
band=2.4ghz-b/g
[admin@MikroTik] interface wireless> monitor wlan1
               status: connected-to-ess
                 band: 2.4ghz-g
            frequency: 2432MHz
              tx-rate: 36Mbps
              rx-rate: 36Mbps
                 ssid: "mmt"
                bssid: 00:0B:6B:34:5A:91
           radio-name: "000B6B345A91"
      signal-strength: -77dBm
   tx-signal-strength: -76dBm
               tx-ccq: 21%
               rx-ccq: 21%
  current-ack-timeout: 56
     current-distance: 56
             wds-link: no
              nstreme: no
        framing-mode: none
     routeros-version: "2.9beta16"
              last-ip: 25.25.25.2
    current-tx-powers: 1Mbps:28,2Mbps:28,5.5Mbps:28,11Mbps:28,6Mbps:27,
                       9Mbps:27,12Mbps:27,18Mbps:27,24Mbps:27,36Mbps:26,
                       48Mbps:25,54Mbps:24

[admin@MikroTik] interface wireless>
```

The 'ess' stands for Extended Service Set (IEEE 802.11 wireless networking).

Nstreme Settings

Home menu level: ***/interface wireless nstreme***

Description

You can switch a wireless card to the nstreme mode. In that case the card will work only with nstreme clients.

Property Description

enable-nstreme (yes | no; default: **no**) - whether to switch the card into the nstreme mode

enable-polling (yes | no; default: **yes**) - whether to use polling for clients

framer-limit (*integer*; default: **3200**) - maximal frame size

framer-policy (*none* | *best-fit* | *exact-size* | *dynamic-size*; default: **none**) - the method how to combine frames (like fast-frames setting in interface configuration). A number of frames may be combined into a bigger one to reduce the amount of protocol overhead (and thus increase speed). The card is not waiting for frames, but in case a number of packets are queued for transmitting, they can be combined. There are several methods of framing:

- **none** - do nothing special, do not combine packets
- **best-fit** - put as much packets as possible in one frame, until the framer-limit limit is met, but do not fragment packets
- **exact-size** - put as much packets as possible in one frame, until the framer-limit limit is met, even if fragmentation will be needed (best performance)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **dynamic-size** - choose the best frame size dynamically

name (*name*) - reference name of the interface

Notes

Such settings as **enable-polling**, **framer-policy** and **framer-limit** are relevant only on Access Point, they are ignored for client devices! The client automatically adapts to AP settings.

Example

To enable the nstreme protocol on the **wlan1** radio with exact-size framing:

```
[admin@MikroTik] interface wireless nstreme> print
 0 name="wlan1" enable-nstreme=no enable-polling=yes framer-policy=none
   framer-limit=3200
[admin@MikroTik] interface wireless nstreme> set wlan1 enable-nstreme=yes \
\... framer-policy=exact-size
```

Nstreme2 Group Settings

Home menu level: ***/interface wireless nstreme-dual***

Description

Two radios in **nstreme-dual-slave** mode can be grouped together to make nstreme2 Point-to-Point connection. To put wireless interfaces into a nstreme2 group, you should set their **mode** to **nstreme-dual-slave**. Many parameters from **/interface wireless** menu are ignored, using the nstreme2, except:

- frequency-mode
- country
- antenna-gain
- tx-power
- tx-power-mode
- antenna-mode

Property Description

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol setting

disable-running-check (yes | no) - whether the interface should always be treated as running even if there is no connection to a remote peer

framer-limit (*integer*; default: **4000**) - maximal frame size

framer-policy (*none* | *best-fit* | *exact-size*; default: **none**) - the method how to combine frames (like fast-frames setting in interface configuration). A number of frames may be combined into one bigger one to reduce the amout of protocol overhead (and thus increase speed). The card are not

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

waiting for frames, but in case a number packets are queued for transmitting, they can be combined. There are several methods of framing:

- **none** - do nothing special, do not combine packets
- **best-fit** - put as much packets as possible in one frame, until the framer-limit limit is met, but do not fragment packets
- **exact-size** - put as much packets as possible in one frame, until the framer-limit limit is met, even if fragmentation will be needed (best performance)

mac-address (*read-only: MAC address*) - MAC address of the receiving wireless card in the set

mtu (*integer*: 0..1600; default: **1500**) - Maximum Transmission Unit

name (*name*) - reference name of the interface

rates-a/g (*multiple choice: 6Mbps, 9Mbps, 12Mbps, 18Mbps, 24Mbps, 36Mbps, 48Mbps, 54Mbps*) - rates to be supported in 802.11a or 802.11g standard

rates-b (*multiple choice: 1Mbps, 2Mbps, 5.5Mbps, 11Mbps*) - rates to be supported in 802.11b standard

remote-mac (*MAC address*; default: **00:00:00:00:00:00**) - which MAC address to connect to (this would be the remote receiver card's MAC address)

rx-band - operating band of the receiving radio

- **2.4ghz-b** - IEEE 802.11b
- **2.4ghz-g** - IEEE 802.11g
- **2.4ghz-g-turbo** - IEEE 802.11g in Atheros proprietary turbo mode (up to 108Mbit)
- **5ghz** - IEEE 802.11a up to 54 Mbit
- **5ghz-turbo** - IEEE 802.11a in Atheros proprietary turbo mode (up to 108Mbit)

rx-frequency (*integer*; default: **5320**) - Frequency to use for receiving frames

rx-radio (*name*) - which radio should be used for receiving frames

tx-band - operating band of the transmitting radio

- **2.4ghz-b** - IEEE 802.11b
- **2.4ghz-g** - IEEE 802.11g
- **2.4ghz-g-turbo** - IEEE 802.11g in Atheros proprietary turbo mode (up to 108Mbit)
- **5ghz** - IEEE 802.11a up to 54 Mbit
- **5ghz-turbo** - IEEE 802.11a in Atheros proprietary turbo mode (up to 108Mbit)

tx-frequency (*integer*; default: **5180**) - Frequency to use for transmitting frames

tx-radio (*name*) - which radio should be used for transmitting frames

Notes

WDS cannot be used on Nstreme-dual links.

The difference between **tx-freq** and **rx-freq** should be about 200MHz (more is recommended) because of the interference that may occur!

You can use different bands for rx and tx links. For example, transmit in **2.4ghz-g-turbo** and receive data, using **2.4ghz-b** band.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

To enable the nstreme2 protocol on a router:

1. Having two Atheros AR5212 based cards which are not used for anything else, to group them into a nstreme interface, switch both of them into **nstreme-slave** mode:

```
[admin@MikroTik] interface wireless> print
Flags: X - disabled, R - running
 0    name="wlan1" mtu=1500 mac-address=00:0B:6B:31:02:4F arp=enabled
      disable-running-check=no interface-type=Atheros AR5212
      radio-name="000B6B31024F" mode=station ssid="MikroTik" frequency=5180
      band=5GHz scan-list=default-ism
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default noise-floor-threshold=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes hide-ssid=no 802.1x-mode=none

 1    name="wlan2" mtu=1500 mac-address=00:0B:6B:30:B4:A4 arp=enabled
      disable-running-check=no interface-type=Atheros AR5212
      radio-name="000B6B30B4A4" mode=station ssid="MikroTik" frequency=5180
      band=5GHz scan-list=default-ism
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default noise-floor-threshold=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes hide-ssid=no 802.1x-mode=none

[admin@MikroTik] interface wireless> set 0,1 mode=nstreme-dual-slave
```

2. Then add nstreme2 interface with exact-size framing:

```
[admin@MikroTik] interface wireless nstreme-dual> add \
\... framer-policy=exact-size
```

3. Configure which card will be receiving and which - transmitting and specify remote receiver card's MAC address:

```
[admin@MikroTik] interface wireless nstreme-dual> print
Flags: X - disabled, R - running
 0 X  name="n-streme1" mtu=1500 mac-address=00:00:00:00:00:00 arp=enabled
      disable-running-check=no tx-radio=(unknown) rx-radio=(unknown)
      remote-mac=00:00:00:00:00:00 tx-band=5GHz tx-frequency=5180
      rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,54Mbps
      rx-band=5GHz rx-frequency=5320 framer-policy=exact-size
      framer-limit=4000

[admin@MikroTik] interface wireless nstreme-dual> set 0 disabled=no \
\... tx-radio=wlan1 rx-radio=wlan2 remote-mac=00:0C:42:05:0B:12
[admin@MikroTik] interface wireless nstreme-dual> print
Flags: X - disabled, R - running
 0 X  name="n-streme1" mtu=1500 mac-address=00:0B:6B:30:B4:A4  arp=enabled
      disable-running-check=no tx-radio=wlan1 rx-radio=wlan2
      remote-mac=00:0C:42:05:0B:12 tx-band=5GHz tx-frequency=5180
      rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,54Mbps
rx-band=5GHz rx-frequency=5320 framer-policy=exact-size
framer-limit=4000
```

Registration Table

Home menu level: ***/interface wireless registration-table***

Description

In the registration table you can see various information about currently connected clients. It is used only for Access Points.

Property Description

ap (*read-only: no* | *yes*) - whether the connected device is an Access Point or not

bytes (*read-only: integer, integer*) - number of sent and received packet bytes

frames (*read-only: integer, integer*) - number of sent and received 802.11 data frames excluding retransmitted data frames

frame-bytes (*read-only: integer, integer*) - number of sent and received data bytes excluding header information

framing-current-size (*read-only: integer*) - current size of combined frames

framing-limit (*read-only: integer*) - maximal size of combined frames

framing-mode (*read-only: none* | *best-fit* | *exact-size*; default: **none**) - the method how to combine frames

hw-frames (*read-only: integer, integer*) - number of sent and received 802.11 data frames including retransmitted data frames

hw-frame-bytes (*read-only: integer, integer*) - number of sent and received data bytes including header information

interface (*read-only: name*) - interface that client is registered to

last-activity (*read-only: time*) - last interface data tx/rx activity

last-ip (*read-only: IP address*) - IP address found in the last IP packet received from the registered client

mac-address (*read-only: MAC address*) - MAC address of the registered client

packets (*read-only: integer, integer*) - number of sent and received network layer packets

packing-size (*read-only: integer*) - maximum packet size in bytes

parent (*read-only: MAC address*) - parent access point's MAC address, if forwarded from another access point

routeros-version (*read-only: name*) - RouterOS version of the registered client

rx-ccq (*read-only: integer*: 0..100) - Client Connection Quality - a value in percent that shows how effective the receive bandwidth is used regarding the theoretically maximum available bandwidth

rx-packed (*read-only: integer*) - number of received packets in form of received-packets/number of packets, which were packed into a larger ones, using fast-frames

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

rx-rate (*read-only: integer*) - receive data rate

signal-strength (*read-only: integer*) - average signal level

tx-ccq (*read-only: integer*: 0..100) - Client Connection Quality - a value in percent that shows how effective the transmit bandwidth is used regarding the theoretically maximum available bandwidth

tx-packed (*read-only: integer*) - number of sent packets in form of sent-packets/number of packets, which were packed into a larger ones, using fast-frames

tx-rate (*read-only: integer*) - transmit data rate

tx-signal-strength (*read-only: integer*) - transmit signal level

type (*read-only: name*) - type of the client

uptime (*read-only: time*) - time the client is associated with the access point

wds (*read-only: no | yes*) - whether the connected client is using wds or not

Example

To see registration table showing all clients currently associated with the access point:

```
[admin@MikroTik] interface wireless registration-table> print
 # INTERFACE RADIO-NAME       MAC-ADDRESS       AP  SIGNAL... TX-RATE
 0 wireless1 000124705304     00:01:24:70:53:04 no  -38dBm... 9Mbps
[admin@MikroTik] interface wireless registration-table>
```

To get additional statistics:

```
[admin@MikroTik] interface wireless> registration-table print stats
0 interface=dfaewad radio-name="000C42050436" mac-address=00:0C:42:05:04:36
  ap=yes wds=no rx-rate=54Mbps tx-rate=54Mbps packets=597,668
  bytes=48693,44191 frames=597,673 frame-bytes=48693,44266 hw-frames=597,683
  hw-frame-bytes=63021,60698 uptime=45m28s last-activity=0s
  signal-strength=-66dBm@54Mbps
  strength-at-rates=-59dBm@1Mbps 13s120ms,-61dBm@6Mbps 7s770ms,-61dBm@9Mbps
                    40m43s970ms,-60dBm@12Mbps 40m43s760ms,-61dBm@18Mbps
                    40m43s330ms,-60dBm@24Mbps 40m43s,-61dBm@36Mbps
                    33m10s230ms,-62dBm@48Mbps 33m9s760ms,-66dBm@54Mbps 10ms
  tx-signal-strength=-65dBm tx-ccq=24% rx-ccq=20% ack-timeout=28 distance=28
  nstreme=no framing-mode=none routeros-version="2.9rc5"
  last-ip=192.168.63.8
  [admin@MikroTik] interface wireless>
```

Connect List

Home menu level: ***/interface wireless connect-list***

Description

The Connect List is a list of rules (order is important), that determine to which AP the station should connect to.

At first, the station is searching for APs all frequencies (from **scan-list**) in the respective band and makes a list of Access Points. If the **ssid** is set under **/interface wireless**, the router removes all Access Points from its AP list which do not have such **ssid**

If a rule is matched and the parameter **connect** is set to **yes**, the station will connect to this AP. If the parameter says **connect=no** or the rule is not matched, we jump to the next rule.

If we have gone through all rules and haven't connected to any AP, yet. The router chooses an AP

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

with the best signal and **ssid** that is set under **/interface wireless**.

In case when the station has not connected to any AP, this process repeats from beginning.

Property Description

area-prefix (*text*) - a string that indicates the beginning from the area string of the AP. If the AP's area begins with area-prefix, then this parameter returns true

connect (yes | no) - whether to connect to AP that matches this rule

interface (*name*) - name of the wireless interface

mac-address (*MAC address*) - MAC address of the AP. If set to 00:00:00:00:00:00, all APs are accepted

min-signal-strength (*integer*) - signal strength in dBm. Rule is matched, if the signal from AP is stronger than this

security-profile (*name*; default: **none**) - name of the security profile, used to connect to the AP. If none, then those security profile is used which is configured for the respective interface

ssid (*text*) - the ssid of the AP. If none set, all ssid's are accepted. Different ssids will be meaningful, if the ssid for the respective interface is set to ""

Access List

Home menu level: ***/interface wireless access-list***

Description

The access list is used by the Access Point to restrict associations of clients. This list contains MAC addresses of clients and determines what action to take when client attempts to connect. Also, the forwarding of frames sent by the client is controlled.

The association procedure is as follows: when a new client wants to associate to the AP that is configured on interface **wlanN**, an entry with client's MAC address and interface **wlanN** is looked up in the access-list. If such entry is found, action specified in the access list is performed, else **default-authentication** and **default-forwarding** arguments of interface **wlanN** are taken.

Property Description

ap-tx-limit (*integer*; default: **0**) - limits data rate for this wireless client (in bps)

- **0** - no limits

authentication (*yes* | *no*; default: **yes**) - whether to accept or to reject this client when it tries to connect

client-tx-limit (*integer*; default: **0**) - limits this client's transmit data rate (in bps). Works only if the client is also a MikroTik Router

- **0** - no limits

forwarding (*yes* | *no*; default: **yes**) - whether to forward the client's frames to other wireless clients

interface (*name*) - name of the respective interface

mac-address (*MAC address*) - MAC address of the client

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

private-algo (*104bit-wep* | *40bit-wep* | *none*) - which encryption algorithm to use

private-key (*text*; default: "") - private key of the client. Used for private-algo

skip-802.1x (yes | no) - not implemented, yet

Notes

If you have default authentication action for the interface set to yes, you can disallow this node to register at the AP's interface wlanN by setting authentication=no for it. Thus, all nodes except this one will be able to register to the interface wlanN.

If you have default authentication action for the interface set to no, you can allow this node to register at the AP's interface wlanN by setting authentication=yes for it. Thus, only the specified nodes will be able to register to the interface wlanN.

Example

To allow authentication and forwarding for the client 00:01:24:70:3A:BB from the wlan1 interface using WEP 40bit algorithm with the key **1234567890**:

```
[admin@MikroTik] interface wireless access-list> add mac-address= \
\... 00:01:24:70:3A:BB interface=wlan1 private-algo=40bit-wep private-key=1234567890
[admin@MikroTik] interface wireless access-list> print
Flags: X - disabled
 0   mac-address=00:01:24:70:3A:BB interface=wlan1 authentication=yes
     forwarding=yes ap-tx-limit=0 client-tx-limit=0 private-algo=40bit-wep
     private-key="1234567890"
[admin@MikroTik] interface wireless access-list>
```

Info

Home menu level: ***/interface wireless info***

Description

This facility provides you with general wireless interface information.

Property Description

2ghz-b-channels (*multiple choice, read-only: 2312, 2317, 2322, 2327, 2332, 2337, 2342, 2347, 2352, 2357, 2362, 2367, 2372, 2412, 2417, 2422, 2427, 2432, 2437, 2442, 2447, 2452, 2457, 2462, 2467, 2472, 2484, 2512, 2532, 2552, 2572, 2592, 2612, 2632, 2652, 2672, 2692, 2712, 2732*) - the list of 2GHz IEEE 802.11b channels (frequencies are given in MHz)

2ghz-g-channels (*multiple choice, read-only: 2312, 2317, 2322, 2327, 2332, 2337, 2342, 2347, 2352, 2357, 2362, 2367, 2372, 2412, 2417, 2422, 2427, 2432, 2437, 2442, 2447, 2452, 2457, 2462, 2467, 2472, 2512, 2532, 2552, 2572, 2592, 2612, 2632, 2652, 2672, 2692, 2712, 2732, 2484*) - the list of 2GHz IEEE 802.11g channels (frequencies are given in MHz)

5ghz-channels (*multiple choice, read-only: 4920, 4925, 4930, 4935, 4940, 4945, 4950, 4955, 4960, 4965, 4970, 4975, 4980, 4985, 4990, 4995, 5000, 5005, 5010, 5015, 5020, 5025, 5030, 5035, 5040, 5045, 5050, 5055, 5060, 5065, 5070, 5075, 5080, 5085, 5090, 5095, 5100, 5105, 5110, 5115, 5120, 5125, 5130, 5135, 5140, 5145, 5150, 5155, 5160, 5165, 5170, 5175, 5180, 5185, 5190, 5195, 5200, 5205, 5210, 5215, 5220, 5225, 5230, 5235, 5240, 5245, 5250, 5255, 5260, 5265, 5270, 5275, 5280,*

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

5285, 5290, 5295, 5300, 5305, 5310, 5315, 5320, 5325, 5330, 5335, 5340, 5345, 5350, 5355, 5360, 5365, 5370, 5375, 5380, 5385, 5390, 5395, 5400, 5405, 5410, 5415, 5420, 5425, 5430, 5435, 5440, 5445, 5450, 5455, 5460, 5465, 5470, 5475, 5480, 5485, 5490, 5495, 5500, 5505, 5510, 5515, 5520, 5525, 5530, 5535, 5540, 5545, 5550, 5555, 5560, 5565, 5570, 5575, 5580, 5585, 5590, 5595, 5600, 5605, 5610, 5615, 5620, 5625, 5630, 5635, 5640, 5645, 5650, 5655, 5660, 5665, 5670, 5675, 5680, 5685, 5690, 5695, 5700, 5705, 5710, 5715, 5720, 5725, 5730, 5735, 5740, 5745, 5750, 5755, 5760, 5765, 5770, 5775, 5780, 5785, 5790, 5795, 5800, 5805, 5810, 5815, 5820, 5825, 5830, 5835, 5840, 5845, 5850, 5855, 5860, 5865, 5870, 5875, 5880, 5885, 5890, 5895, 5900, 5905, 5910, 5915, 5920, 5925, 5930, 5935, 5940, 5945, 5950, 5955, 5960, 5965, 5970, 5975, 5980, 5985, 5990, 5995, 6000, 6005, 6010, 6015, 6020, 6025, 6030, 6035, 6040, 6045, 6050, 6055, 6060, 6065, 6070, 6075, 6080, 6085, 6090, 6095, 6100) - the list of 5GHz channels (frequencies are given in MHz)

5ghz-turbo-channels (*multiple choice, read-only: 4920, 4925, 4930, 4935, 4940, 4945, 4950, 4955, 4960, 4965, 4970, 4975, 4980, 4985, 4990, 4995, 5000, 5005, 5010, 5015, 5020, 5025, 5030, 5035, 5040, 5045, 5050, 5055, 5060, 5065, 5070, 5075, 5080, 5085, 5090, 5095, 5100, 5105, 5110, 5115, 5120, 5125, 5130, 5135, 5140, 5145, 5150, 5155, 5160, 5165, 5170, 5175, 5180, 5185, 5190, 5195, 5200, 5205, 5210, 5215, 5220, 5225, 5230, 5235, 5240, 5245, 5250, 5255, 5260, 5265, 5270, 5275, 5280, 5285, 5290, 5295, 5300, 5305, 5310, 5315, 5320, 5325, 5330, 5335, 5340, 5345, 5350, 5355, 5360, 5365, 5370, 5375, 5380, 5385, 5390, 5395, 5400, 5405, 5410, 5415, 5420, 5425, 5430, 5435, 5440, 5445, 5450, 5455, 5460, 5465, 5470, 5475, 5480, 5485, 5490, 5495, 5500, 5505, 5510, 5515, 5520, 5525, 5530, 5535, 5540, 5545, 5550, 5555, 5560, 5565, 5570, 5575, 5580, 5585, 5590, 5595, 5600, 5605, 5610, 5615, 5620, 5625, 5630, 5635, 5640, 5645, 5650, 5655, 5660, 5665, 5670, 5675, 5680, 5685, 5690, 5695, 5700, 5705, 5710, 5715, 5720, 5725, 5730, 5735, 5740, 5745, 5750, 5755, 5760, 5765, 5770, 5775, 5780, 5785, 5790, 5795, 5800, 5805, 5810, 5815, 5820, 5825, 5830, 5835, 5840, 5845, 5850, 5855, 5860, 5865, 5870, 5875, 5880, 5885, 5890, 5895, 5900, 5905, 5910, 5915, 5920, 5925, 5930, 5935, 5940, 5945, 5950, 5955, 5960, 5965, 5970, 5975, 5980, 5985, 5990, 5995, 6000, 6005, 6010, 6015, 6020, 6025, 6030, 6035, 6040, 6045, 6050, 6055, 6060, 6065, 6070, 6075, 6080, 6085, 6090, 6095, 6100*) - the list of 5GHz-turbo channels (frequencies are given in MHz)

ack-timeout-control (*read-only: yes* | *no*) - provides information whether this device supports transmission acceptance timeout control

alignment-mode (*read-only: yes* | *no*) - is the alignment-only mode supported by this interface

burst-support (yes | no) - whether the interface supports data bursts (burst-time)

chip-info (*read-only: text*) - information from EEPROM

default-periodic-calibration (*read-only:* yes | no) - whether the card supports periodic-calibration

firmware (*read-only: text*) - current firmware of the interface (used only for Prism chipset based cards)

interface-type (*read-only: text*) - shows the hardware interface type

noise-floor-control (*read-only: yes* | *no*) - does this interface support noise-floor-thershold detection

nstreme-support (*read-only:* yes | no) - whether the card supports n-streme protocol

scan-support (yes | no) - whether the interface supports scan function ('/interface wireless scan')

supported-bands (*multiple choice, read-only: 2ghz-b, 5ghz, 5ghz-turbo, 2ghz-g*) - the list of supported bands

tx-power-control (*read-only: yes* | *no*) - provides information whether this device supports transmission power control

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

virtual-aps (*read-only: yes | no*) - whether this interface supports Virtual Access Points ('/interface wireless add')

Notes

There is a special argument for the print command - print count-only. It forces the print command to print only the count of information topics.

/interface wireless info print command shows only channels supported by a particular card.

Example

```
[admin@MikroTik] interface wireless info> print
 0 interface-type=Atheros AR5413
   chip-info="mac:0xa/0x5, phy:0x61, a5:0x63, a2:0x0, eeprom:0x5002"
   tx-power-control=yes ack-timeout-control=yes alignment-mode=yes
   virtual-aps=yes noise-floor-control=yes scan-support=yes burst-support=yes
   nstreme-support=yes default-periodic-calibration=enabled
   supported-bands=2ghz-b,5ghz,5ghz-turbo,2ghz-g,2ghz-g-turbo
   2ghz-b-channels=2312:0,2317:0,2322:0,2327:0,2332:0,2337:0,2342:0,2347:0,
                   2352:0,2357:0,2362:0,2367:0,2372:0,2377:0,2382:0,2387:0,
                   2392:0,2397:0,2402:0,2407:0,2412:0,2417:0,2422:0,2427:0,
                   2432:0,2437:0,2442:0,2447:0,2452:0,2457:0,2462:0,2467:0,
                   2472:0,2477:0,2482:0,2487:0,2492:0,2497:0,2314:0,2319:0,
                   2324:0,2329:0,2334:0,2339:0,2344:0,2349:0,2354:0,2359:0,
                   2364:0,2369:0,2374:0,2379:0,2384:0,2389:0,2394:0,2399:0,
                   2404:0,2409:0,2414:0,2419:0,2424:0,2429:0,2434:0,2439:0,
                   2444:0,2449:0,2454:0,2459:0,2464:0,2469:0,2474:0,2479:0,
                   2484:0,2489:0,2494:0,2499:0
   5ghz-channels=4920:0,4925:0,4930:0,4935:0,4940:0,4945:0,4950:0,4955:0,
                 4960:0,4965:0,4970:0,4975:0,4980:0,4985:0,4990:0,4995:0,
                 5000:0,5005:0,5010:0,5015:0,5020:0,5025:0,5030:0,5035:0,
                 5040:0,5045:0,5050:0,5055:0,5060:0,5065:0,5070:0,5075:0,
                 5080:0,5085:0,5090:0,5095:0,5100:0,5105:0,5110:0,5115:0,
                 5120:0,5125:0,5130:0,5135:0,5140:0,5145:0,5150:0,5155:0,
                 5160:0,5165:0,5170:0,5175:0,5180:0,5185:0,5190:0,5195:0,
                 5200:0,5205:0,5210:0,5215:0,5220:0,5225:0,5230:0,5235:0,
                 5240:0,5245:0,5250:0,5255:0,5260:0,5265:0,5270:0,5275:0,
                 5280:0,5285:0,5290:0,5295:0,5300:0,5305:0,5310:0,5315:0,
                 5320:0,5325:0,5330:0,5335:0,5340:0,5345:0,5350:0,5355:0,
                 5360:0,5365:0,5370:0,5375:0,5380:0,5385:0,5390:0,5395:0,
                 5400:0,5405:0,5410:0,5415:0,5420:0,5425:0,5430:0,5435:0,
                 5440:0,5445:0,5450:0,5455:0,5460:0,5465:0,5470:0,5475:0,
                 5480:0,5485:0,5490:0,5495:0,5500:0,5505:0,5510:0,5515:0,
                 5520:0,5525:0,5530:0,5535:0,5540:0,5545:0,5550:0,5555:0,
                 5560:0,5565:0,5570:0,5575:0,5580:0,5585:0,5590:0,5595:0,
                 5600:0,5605:0,5610:0,5615:0,5620:0,5625:0,5630:0,5635:0,
                 5640:0,5645:0,5650:0,5655:0,5660:0,5665:0,5670:0,5675:0,
                 5680:0,5685:0,5690:0,5695:0,5700:0,5705:0,5710:0,5715:0,
                 5720:0,5725:0,5730:0,5735:0,5740:0,5745:0,5750:0,5755:0,
                 5760:0,5765:0,5770:0,5775:0,5780:0,5785:0,5790:0,5795:0,
                 5800:0,5805:0,5810:0,5815:0,5820:0,5825:0,5830:0,5835:0,
                 5840:0,5845:0,5850:0,5855:0,5860:0,5865:0,5870:0,5875:0,
                 5880:0,5885:0,5890:0,5895:0,5900:0,5905:0,5910:0,5915:0,
                 5920:0,5925:0,5930:0,5935:0,5940:0,5945:0,5950:0,5955:0,
                 5960:0,5965:0,5970:0,5975:0,5980:0,5985:0,5990:0,5995:0,
                 6000:0,6005:0,6010:0,6015:0,6020:0,6025:0,6030:0,6035:0,
                 6040:0,6045:0,6050:0,6055:0,6060:0,6065:0,6070:0,6075:0,
                 6080:0,6085:0,6090:0,6095:0,6100:0
   5ghz-turbo-channels=4920:0,4925:0,4930:0,4935:0,4940:0,4945:0,4950:0,4955:0,
                       4960:0,4965:0,4970:0,4975:0,4980:0,4985:0,4990:0,4995:0,
                       5000:0,5005:0,5010:0,5015:0,5020:0,5025:0,5030:0,5035:0,
                       5040:0,5045:0,5050:0,5055:0,5060:0,5065:0,5070:0,5075:0,
                       5080:0,5085:0,5090:0,5095:0,5100:0,5105:0,5110:0,5115:0,
                       5120:0,5125:0,5130:0,5135:0,5140:0,5145:0,5150:0,5155:0,
                       5160:0,5165:0,5170:0,5175:0,5180:0,5185:0,5190:0,5195:0,
                       5200:0,5205:0,5210:0,5215:0,5220:0,5225:0,5230:0,5235:0,
                       5240:0,5245:0,5250:0,5255:0,5260:0,5265:0,5270:0,5275:0,
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
                        5280:0,5285:0,5290:0,5295:0,5300:0,5305:0,5310:0,5315:0,
                        5320:0,5325:0,5330:0,5335:0,5340:0,5345:0,5350:0,5355:0,
                        5360:0,5365:0,5370:0,5375:0,5380:0,5385:0,5390:0,5395:0,
                        5400:0,5405:0,5410:0,5415:0,5420:0,5425:0,5430:0,5435:0,
                        5440:0,5445:0,5450:0,5455:0,5460:0,5465:0,5470:0,5475:0,
                        5480:0,5485:0,5490:0,5495:0,5500:0,5505:0,5510:0,5515:0,
                        5520:0,5525:0,5530:0,5535:0,5540:0,5545:0,5550:0,5555:0,
                        5560:0,5565:0,5570:0,5575:0,5580:0,5585:0,5590:0,5595:0,
                        5600:0,5605:0,5610:0,5615:0,5620:0,5625:0,5630:0,5635:0,
                        5640:0,5645:0,5650:0,5655:0,5660:0,5665:0,5670:0,5675:0,
                        5680:0,5685:0,5690:0,5695:0,5700:0,5705:0,5710:0,5715:0,
                        5720:0,5725:0,5730:0,5735:0,5740:0,5745:0,5750:0,5755:0,
                        5760:0,5765:0,5770:0,5775:0,5780:0,5785:0,5790:0,5795:0,
                        5800:0,5805:0,5810:0,5815:0,5820:0,5825:0,5830:0,5835:0,
                        5840:0,5845:0,5850:0,5855:0,5860:0,5865:0,5870:0,5875:0,
                        5880:0,5885:0,5890:0,5895:0,5900:0,5905:0,5910:0,5915:0,
                        5920:0,5925:0,5930:0,5935:0,5940:0,5945:0,5950:0,5955:0,
                        5960:0,5965:0,5970:0,5975:0,5980:0,5985:0,5990:0,5995:0,
                        6000:0,6005:0,6010:0,6015:0,6020:0,6025:0,6030:0,6035:0,
                        6040:0,6045:0,6050:0,6055:0,6060:0,6065:0,6070:0,6075:0,
                        6080:0,6085:0,6090:0,6095:0,6100:0
    2ghz-g-channels=2312:0,2317:0,2322:0,2327:0,2332:0,2337:0,2342:0,2347:0,
                    2352:0,2357:0,2362:0,2367:0,2372:0,2377:0,2382:0,2387:0,
                    2392:0,2397:0,2402:0,2407:0,2412:0,2417:0,2422:0,2427:0,
                    2432:0,2437:0,2442:0,2447:0,2452:0,2457:0,2462:0,2467:0,
                    2472:0,2477:0,2482:0,2487:0,2492:0,2497:0,2314:0,2319:0,
                    2324:0,2329:0,2334:0,2339:0,2344:0,2349:0,2354:0,2359:0,
                    2364:0,2369:0,2374:0,2379:0,2384:0,2389:0,2394:0,2399:0,
                    2404:0,2409:0,2414:0,2419:0,2424:0,2429:0,2434:0,2439:0,
                    2444:0,2449:0,2454:0,2459:0,2464:0,2469:0,2474:0,2479:0,
                    2484:0,2489:0,2494:0,2499:0
    2ghz-g-turbo-channels=2312:0,2317:0,2322:0,2327:0,2332:0,2337:0,2342:0,
                          2347:0,2352:0,2357:0,2362:0,2367:0,2372:0,2377:0,
                          2382:0,2387:0,2392:0,2397:0,2402:0,2407:0,2412:0,
                          2417:0,2422:0,2427:0,2432:0,2437:0,2442:0,2447:0,
                          2452:0,2457:0,2462:0,2467:0,2472:0,2477:0,2482:0,
                          2487:0,2492:0,2497:0,2314:0,2319:0,2324:0,2329:0,
                          2334:0,2339:0,2344:0,2349:0,2354:0,2359:0,2364:0,
                          2369:0,2374:0,2379:0,2384:0,2389:0,2394:0,2399:0,
                          2404:0,2409:0,2414:0,2419:0,2424:0,2429:0,2434:0,
                          2439:0,2444:0,2449:0,2454:0,2459:0,2464:0,2469:0,
                          2474:0,2479:0,2484:0,2489:0,2494:0,2499:0
[admin@MikroTik] interface wireless>
```

Virtual Access Point Interface

Home menu level: ***/interface wireless***

Description

Virtual Access Point (VAP) interface is used to have an additional AP. You can create a new AP with different **ssid** and **mac-address**. It can be compared with a VLAN where the **ssid** from VAP is the VLAN **tag** and the hardware interface is the VLAN switch.

You can add up to 7 VAP interfaces for each hardware interface.

RouterOS supports VAP feature for Atheros AR5212 and newer.

Property Description

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*) - ARP mode

default-authentication (*yes* | *no*; default: **yes**) - whether to accept or reject a client that wants to associate, but is not in the access-list

default-forwarding (*yes* | *no*; default: **yes**) - whether to forward frames to other AP clients or not

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

disabled (*yes* | *no*; default: **yes**) - whether to disable the interface or not

disable-running-check (*yes* | *no*; default: **no**) - disable running check. For 'broken' cards it is a good idea to set this value to 'yes'

hide-ssid (*yes* | *no*; default: **no**) - whether to hide ssid or not in the beacon frames:

- **yes** - ssid is not included in the beacon frames. AP replies only to probe-requests with the given ssid
- **no** - ssid is included in beacon frames. AP replies to probe-requests with the given ssid and to 'broadcast ssid'

mac-address (*MAC address*; default: **02:00:00:AA:00:00**) - MAC address of VAP. You can define your own value for mac-address

master-interface (*name*) - hardware interface to use for VAP

max-station-count (*integer*; default: **2007**) - number of clients that can connect to this AP simultaneously

mtu (*integer*: 68..1600; default: **1500**) - Maximum Transmission Unit

name (*name*; default: **wlanN**) - interface name

ssid (*text*; default: **MikroTik**) - the service set identifier

WDS Interface Configuration

Home menu level: ***/interface wireless wds***

Description

WDS (Wireless Distribution System) allows packets to pass from one wireless AP (Access Point) to another, just as if the APs were ports on a wired Ethernet switch. APs must use the same standard (802.11a, 802.11b or 802.11g) and work on the same frequencies in order to connect to each other.

There are two possibilities to create a WDS interface:

- **dynamic** - is created 'on the fly' and appers under wds menu as a dynamic interface
- **static** - is created manually

Property Description

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol

- **disabled** - the interface will not use ARP
- **enabled** - the interface will use ARP
- **proxy-arp** - the interface will use the ARP proxy feature
- **reply-only** - the interface will only reply to the requests originated to its own IP addresses. Neighbour MAC addresses will be resolved using /ip arp statically set table only

disable-running-check (*yes* | *no*; default: **no**) - disable running check. For 'broken' wireless cards it is a good idea to set this value to 'yes'

mac-address (*read-only: MAC address*; default: **00:00:00:00:00:00**) - MAC address of the master-interface. Specifying master-interface, this value will be set automatically

master-interface (*name*) - wireless interface which will be used by WDS

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

mtu (*integer*: 0..65336; default: **1500**) - Maximum Transmission Unit

name (*name*; default: **wdsN**) - WDS interface name

wds-address (*MAC address*) - MAC address of the remote WDS host

Notes

When the link between WDS devices, using **wds-mode=dynamic**, goes down, the dynamic WDS interfaces disappear and if there are any IP addresses set on this interface, their 'interface' setting will change to **(unknown)**. When the link comes up again, the 'interface' value will not change - it will remain as **(unknown)**. That's why it is not recommended to add IP addresses to dynamic WDS interfaces.

If you want to use dynamic WDS in a bridge, set the **wds-default-bridge** value to desired bridge interface name. When the link will go down and then it comes up, the dynamic WDS interface will be put in the specified bridge automatically.

As the routers which are in WDS mode have to communicate at equal frequencies, it is not recommended to use **WDS** and **DFS** simultaneously - it is most probable that these routers will not connect to each other.

Example

```
[admin@MikroTik] interface wireless wds> add master-interface=wlan1 \
\... wds-address=00:0B:6B:30:2B:27 disabled=no
[admin@MikroTik] interface wireless wds> print
Flags: X - disabled, R - running, D - dynamic
  0  R  name="wds1" mtu=1500 mac-address=00:0B:6B:30:2B:23 arp=enabled
        disable-running-check=no master-inteface=wlan1
        wds-address=00:0B:6B:30:2B:27

[admin@MikroTik] interface wireless wds>
```

Align

Home menu level: ***/interface wireless align***

Description

This feature is created to position wireless links. The **align** submenu describes properties which are used if **/interface wireless mode** is set to **alignment-only**. In this mode the interface 'listens' to those packets which are sent to it from other devices working on the same channel. The interface also can send special packets which contains information about its parameters.

Property Description

active-mode (*yes* | *no*; default: **yes**) - whether the interface will receive and transmit 'alignment' packets or it will only receive them

audio-max (*integer*; default: **-20**) - signal-strength at which audio (beeper) frequency will be the highest

audio-min (*integer*; default: **-100**) - signal-strength at which audio (beeper) frequency will be the lowest

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

audio-monitor (*MAC address*; default: **00:00:00:00:00:00**) - MAC address of the remote host which will be 'listened'

filter-mac (*MAC address*; default: **00:00:00:00:00:00**) - in case if you want to receive packets from only one remote host, you should specify here its MAC address

frame-size (*integer*: 200..1500; default: **300**) - size of 'alignment' packets that will be transmitted

frames-per-second (*integer*: 1..100; default: **25**) - number of frames that will be sent per second (in active-mode)

receive-all (*yes* | *no*; default: **no**) - whether the interface gathers packets about other 802.11 standard packets or it will gather only 'alignment' packets

ssid-all (*yes* | *no*; default: **no**) - whether you want to accept packets from hosts with other ssid than yours

test-audio (*integer*) - test the beeper for 10 seconds

Notes

If you are using the command **/interface wireless align monitor** then it will automatically change the wireless interface's mode from **station**, **bridge** or **ap-bridge** to **alignment-only**.

Example

```
[admin@MikroTik] interface wireless align> print
           frame-size: 300
          active-mode: yes
          receive-all: yes
        audio-monitor: 00:00:00:00:00:00
           filter-mac: 00:00:00:00:00:00
             ssid-all: no
    frames-per-second: 25
            audio-min: -100
            audio-max: -20
[admin@MikroTik] interface wireless align>
```

Align Monitor

Command name: ***/interface wireless align monitor***

Description

This command is used to monitor current signal parameters to/from a remote host.

Property Description

address (*read-only: MAC address*) - MAC address of the remote host

avg-rxq (*read-only: integer*) - average signal strength of received packets since last display update on screen

correct (*read-only: percentage*) - how many undamaged packets were received

last-rx (*read-only: time*) - time in seconds before the last packet was received

last-tx (*read-only: time*) - time in seconds when the last TXQ info was received

rxq (*read-only: integer*) - signal strength of last received packet

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

ssid (*read-only: text*) - service set identifier

txq (*read-only: integer*) - the last received signal strength from our host to the remote one

Example

```
[admin@MikroTik] interface wireless align> monitor wlan2
 # ADDRESS           SSID          RXQ AVG-RXQ LAST-RX TXQ LAST-TX CORRECT
 0 00:01:24:70:4B:FC wirelesa      -60 -60     0.01    -67 0.01    100 %

[admin@MikroTik] interface wireless align>
```

Frequency Monitor

Description

Aproximately shows how loaded are the wireless channels.

Property Description

freq (*read-only: integer*) - shows current channel

use (*read-only: percentage*) - shows usage in current channel

Example

Monitor 802.11b network load:

```
[admin@MikroTik] interface wireless> frequency-monitor wlan1

FREQ          USE
2412MHz       3.8%
2417MHz       9.8%
2422MHz       2%
2427MHz       0.8%
2432MHz       0%
2437MHz       0.9%
2442MHz       0.9%
2447MHz       2.4%
2452MHz       3.9%
2457MHz       7.5%
2462MHz       0.9%
```

To monitor other bands, change the the **band** setting for the respective wireless interface.

Manual Transmit Power Table

Home menu level: ***/interface wireless manual-tx-power-table***

Description

In this submenu you can define signal strength for each rate. You should be aware that you can damage your wireless card if you set higher output power than it is allowed. Note that the values in this table are set in **dBm**! **NOT in mW!** Therefore this table is used mainly to reduce the transmit power of the card.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Property Description

manual-tx-powers (*text*) - define tx-power in dBm for each rate, separate by commas

Example

To set the following transmit powers at each rates: 1Mbps@10dBm, 2Mbps@10dBm, 5.5Mbps@9dBm, 11Mbps@7dBm, do the following:

```
[admin@MikroTik] interface wireless manual-tx-power-table> print
 0 name="wlan1" manual-tx-powers=1Mbps:17,2Mbps:17,5.5Mbps:17,11Mbps:17,6Mbps:17
,
                                9Mbps:17,12Mbps:17,18Mbps:17,24Mbps:17,
                                36Mbps:17,48Mbps:17,54Mbps:17

[admin@MikroTik] interface wireless manual-tx-power-table> set 0 \
   manual-tx-powers=1Mbps:10,2Mbps:10,5.5Mbps:9,11Mbps:7

[admin@MikroTik] interface wireless manual-tx-power-table> print
 0 name="wlan1" manual-tx-powers=1Mbps:10,2Mbps:10,5.5Mbps:9,11Mbps:7
[admin@MikroTik] interface wireless manual-tx-power-table>
```

Network Scan

Command name: ***/interface wireless scan interface_name***

Description

This is a feature that allows you to scan all avaliable wireless networks. While scanning, the card unregisters itself from the access point (in station mode), or unregisters all clients (in bridge or ap-bridge mode). Thus, network connections are lost while scanning.

Property Description

address (*read-only: MAC address*) - MAC address of the AP

band (*read-only: text*) - in which standard does the AP operate

bss (*read-only: yes* | *no*) - basic service set

freeze-time-interval (*time*; default: **1s**) - time in seconds to refresh the displayed data

freq (*read-only: integer*) - the frequency of AP

interface_name (*name*) - the name of interface which will be used for scanning APs

privacy (*read-only: yes* | *no*) - whether all data is encrypted or not

signal-strength (*read-only: integer*) - signal strength in dBm

ssid (*read-only: text*) - service set identifier of the AP

Example

Scan the 5GHz band:

```
[admin@MikroTik] interface wireless> scan wlan1
Flags: A - active, B - bss, P - privacy, R - routeros-network, N - nstreme
      ADDRESS           SSID              BAND       FREQ SIG RADIO-NAME
AB R  00:0C:42:05:00:28 test              5ghz       5180 -77 000C42050028
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
AB R  00:02:6F:20:34:82 aap1              5ghz       5180 -73 00026F203482
AB    00:0B:6B:30:80:0F www               5ghz       5180 -84
AB R  00:0B:6B:31:B6:D7 www               5ghz       5180 -81 000B6B31B6D7
AB R  00:0B:6B:33:1A:D5 R52_test_new      5ghz       5180 -79 000B6B331AD5
AB R  00:0B:6B:33:0D:EA short5            5ghz       5180 -70 000B6B330DEA
AB R  00:0B:6B:31:52:69 MikroTik          5ghz       5220 -69 000B6B315269
AB R  00:0B:6B:33:12:BF long2             5ghz       5260 -55 000B6B3312BF
-- [Q quit|D dump|C-z pause]
[admin@MikroTik] interface wireless>
```

Security Profiles

Home menu level: ***/interface wireless security-profiles***

Description

This section provides WEP (Wired Equivalent Privacy) and WPA (Wi-Fi Protected Access) functions to wireless interfaces.

WPA

The Wi-Fi Protected Access is a combination of 802.1X, EAP, MIC, TKIP and AES. This is a easy to configure and secure wireless mechanism.

WEP

The Wired Equivalent Privacy encrypts data only between 802.11 devices, using static keys. It is not considered as a very secure wireless data encryption mechanism, though it is better than no encryption at all.

The configuration of WEP is quite simple, using MikroTik RouterOS security profiles.

Property Description

group-key-update (*time*; default: **5m**) - how often to update group key. This parameter is used only if the wireless card is configured as an Access Point

mode (*none* | *static-keys-optional* | *static-keys-required* | *wpa-psk*; default: **none**) - security mode:

- **none** - do not encrypt packets and do not accept encrypted packets
- **static-keys-optional** - if there is a static-sta-private-key set, use it. Otherwise, if the interface is set in an AP mode, do not use encryption, if the the interface is in station mode, use encryption if the static-transmit-key is set
- **static-keys-required** - encrypt all packets and accept only encrypted packets
- **wpa-psk** - use WPA Pre-Shared Key mode

name (*name*) - descriptive name for the security profile

pre-shared-key (*text*; default: "") - string, which is used as the WPA Pre Shared Key. It must be the same on AP and station to communicate

radius-mac-authentication (*no* | *yes*; default: **no**) - whether to use Radius server for MAC authentication

static-algo-0 (*none* | *40bit-wep* | *104bit-wep* | *aes-ccm* | *tkip*; default: **none**) - which encryption

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

algorithm to use:

- **none** - do not use encryption and do not accept encrypted packets
- **40bit-wep** - use the 40bit encryption (also known as 64bit-wep) and accept only these packets
- **104bit-wep** - use the 104bit encryption (also known as 128bit-wep) and accept only these packets
- **aes-ccm** - use the AES-CCM (Advanced Encryption Standard in Counter with CBC-MAC) encryption algorithm and accept only these packets
- **tkip** - use the TKIP (Temporal Key Integrity Protocol) and accept only these packets

static-algo-1 (*none* | *40bit-wep* | *104bit-wep* | *aes-ccm* | *tkip*; default: **none**) - which encryption algorithm to use:

- **none** - do not use encryption and do not accept encrypted packets
- **40bit-wep** - use the 40bit encryption (also known as 64bit-wep) and accept only these packets
- **104bit-wep** - use the 104bit encryption (also known as 128bit-wep) and accept only these packets
- **aes-ccm** - use the AES-CCM (Advanced Encryption Standard in Counter with CBC-MAC) encryption algorithm and accept only these packets
- **tkip** - use the TKIP (Temporal Key Integrity Protocol) and accept only these packets

static-algo-2 (*none* | *40bit-wep* | *104bit-wep* | *aes-ccm* | *tkip*; default: **none**) - which encryption algorithm to use:

- **none** - do not use encryption and do not accept encrypted packets
- **40bit-wep** - use the 40bit encryption (also known as 64bit-wep) and accept only these packets
- **104bit-wep** - use the 104bit encryption (also known as 128bit-wep) and accept only these packets
- **aes-ccm** - use the AES-CCM (Advanced Encryption Standard in Counter with CBC-MAC) encryption algorithm and accept only these packets
- **tkip** - use the TKIP (Temporal Key Integrity Protocol) and accept only these packets

static-algo-3 (*none* | *40bit-wep* | *104bit-wep* | *aes-ccm* | *tkip*; default: **none**) - which encryption algorithm to use:

- **none** - do not use encryption and do not accept encrypted packets
- **40bit-wep** - use the 40bit encryption (also known as 64bit-wep) and accept only these packets
- **104bit-wep** - use the 104bit encryption (also known as 128bit-wep) and accept only these packets
- **aes-ccm** - use the AES-CCM (Advanced Encryption Standard in Counter with CBC-MAC) encryption algorithm and accept only these packets
- **tkip** - use the TKIP (Temporal Key Integrity Protocol) and accept only these packets

static-key-0 (*text*) - hexadecimal key which will be used to encrypt packets with the 40bit-wep or 104bit-wep algorithm (algo-0). If AES-CCM is used, the key must consist of even number of characters and must be at least 32 characters long. For TKIP, the key must be at least 64 characters long and also must consist of even number characters

static-key-1 (*text*) - hexadecimal key which will be used to encrypt packets with the 40bit-wep or 104bit-wep algorithm (algo-0). If AES-CCM is used, the key must consist of even number of characters and must be at least 32 characters long. For TKIP, the key must be at least 64 characters

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

long and also must consist of even number characters

static-key-2 (*text*) - hexadecimal key which will be used to encrypt packets with the 40bit-wep or 104bit-wep algorithm (algo-0). If AES-CCM is used, the key must consist of even number of characters and must be at least 32 characters long. For TKIP, the key must be at least 64 characters long and also must consist of even number characters

static-key-3 (*text*) - hexadecimal key which will be used to encrypt packets with the 40bit-wep or 104bit-wep algorithm (algo-0). If AES-CCM is used, the key must consist of even number of characters and must be at least 32 characters long. For TKIP, the key must be at least 64 characters long and also must consist of even number characters

static-sta-private-algo (*none* | *40bit-wep* | *104bit-wep* | *aes-ccm* | *tkip*) - algorithm to use if the static-sta-private-key is set. Used to commumicate between 2 devices

static-sta-private-key (*text*) - if this key is set in station mode, use this key for encryption. In AP mode you have to specify static-private keys in the access-list or use the Radius server using radius-mac-authentication. Used to commumicate between 2 devices

static-transmit-key (*static-key-0* | *static-key-1* | *static-key-2* | *static-key-3*; default: **static-key-0**) - which key to use for broadcast packets. Used in AP mode

wpa-group-ciphers (*aes-ccm* | *tkip*; default: **""**) - which algorithms to use for WPA group communications (for multicast and broadcast packets). If the interface is an Access Point, it will use the "strongest" algorithm from AES and TKIP (AES is "stronger"). If the interface acts as a station, it will connect to Access Points which support at least one of selected algorithms

wpa-unicast-ciphers (*aes-ccm* | *tkip*; default: **""**) - which algorithms are allowed to use for unicast communications. If the interface is an Access Point, then it sends these algorithms as supported. If it is a station, then it will connect only to APs which support any of these algorithms

Notes

The keys used for encryption are in hexadecimal form. If you use **40bit-wep**, the key has to be 10 characters long, if you use **104bit-wep**, the key has to be 26 characters long.

Prism card doesn't report that the use of WEP is required for all data type frames, which means that some clients will not see that access point uses encryption and will not be able to connect to such AP. This is a Prism hardware problem and can not be fixed. Use Atheros-based cards (instead of Prism) on APs if you want to provide WEP in your wireless network.

Sniffer

Home menu level: ***/interface wireless sniffer***

Description

With wireless sniffer you can sniff packets from wireless networks.

Property Description

channel-time (*time*; default: **200ms**) - how long to sniff each channel, if multiple-channels is set to yes

file-limit (*integer*; default: **10**) - limits file-name's file size (measured in kilobytes)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

file-name (*text*; default: **""**) - name of the file where to save packets in PCAP format. If file-name is not defined, packets are not saved into a file

memory-limit (*integer*; default: **1000**) - how much memory to use (in kilobytes) for sniffed packets

multiple-channels (yes | no; default: **no**) - whether to sniff multiple channels or a single channel

- **no** - wireless sniffer sniffs only one channel in frequency that is configured in /interface wireless
- **yes** - sniff in all channels that are listed in the scan-list in /interface wireless

only-headers (yes | no; default: **no**) - sniff only wireless packet heders

receive-errors (yes | no; default: **no**) - whether to receive packets with CRC errors

streaming-enabled (yes | no; default: **no**) - whether to send packets to server in TZSP format

streaming-max-rate (*integer*; default: **0**) - how many packets per second the router will accept

- **0** - no packet per second limitation

streaming-server (*IP address*; default: **0.0.0.0**) - streaming server's IP address

Sniffer Sniff

Home menu level: ***/interface wireless sniffer sniff***

Description

Wireless Sniffer Sniffs packets

Property Description

file-over-limit-packets (*read-only: integer*) - how many packets are dropped because of exceeding file-limit

file-saved-packets (*read-only: integer*) - number of packets saved to file

file-size (*read-only: integer*) - current file size (kB)

memory-over-limit-packets (*read-only: integer*) - number of packets that are dropped because of exceeding memory-limit

memory-saved-packets (*read-only: integer*) - how many packets are stored in mermory

memory-size (*read-only: integer*) - how much memory is currently used for sniffed packets (kB)

processed-packets (*read-only: integer*) - number of sniffed packets

real-file-limit (*read-only: integer*) - the real file size limit. It is calculated from the beginning of sniffing to reserve at least 1MB free space on the disk

real-memory-limit (*read-only: integer*) - the real memory size limit. It is calculated from the beginning of sniffing to reserve at least 1MB of free space in the memory

stream-dropped-packets (*read-only: integer*) - number of packets that are dropped because of exceeding streaming-max-rate

stream-sent-packets (*read-only: integer*) - number of packets that are sent to the streaming server

Command Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

save - saves sniffed packets from the memory to file-name in PCAP format

Sniffer Packets

Description

Wireless Sniffer sniffed packets. If packets Cyclic Redundancy Check (CRC) field detects error, it will be displayed by crc-error flag.

Property Description

dst (*read-only: MAC address*) - the receiver's MAC address

freq (*read-only: integer*) - frequency

interface (*read-only: text*) - wireless interface that captures packets

signal@rate (*read-only: text*) - at which signal-strength and rate was the packet received

src (*read-only: MAC address*) - the sender's MAC address

time (*read-only: time*) - time when the packet was received, starting from the beginning of sniffing

type (*read-only: assoc-req* | *assoc-resp* | *reassoc-req* | *reassoc-resp* | *probe-req* | *probe-resp* | *beacon* | *atim* | *disassoc* | *auth* | *deauth* | *ps-poll* | *rts* | *cts* | *ack* | *cf-end* | *cf-endack* | *data* | *d-cfack* | *d-cfpoll* | *d-cfackpoll* | *data-null* | *nd-cfack* | *nd-cfpoll* | *nd-cfackpoll*) - type of the sniffed packet

Example

Sniffed packets:

```
[admin@MikroTik] interface wireless sniffer packet> pr
Flags: E - crc-error
 #   FREQ SIGNAL@RATE    SRC               DST               TYPE
 0   2412 -73dBm@1Mbps   00:0B:6B:31:00:53 FF:FF:FF:FF:FF:FF beacon
 1   2412 -91dBm@1Mbps   00:02:6F:01:CE:2E FF:FF:FF:FF:FF:FF beacon
 2   2412 -45dBm@1Mbps   00:02:6F:05:68:D3 FF:FF:FF:FF:FF:FF beacon
 3   2412 -72dBm@1Mbps   00:60:B3:8C:98:3F FF:FF:FF:FF:FF:FF beacon
 4   2412 -65dBm@1Mbps   00:01:24:70:3D:4E FF:FF:FF:FF:FF:FF probe-req
 5   2412 -60dBm@1Mbps   00:01:24:70:3D:4E FF:FF:FF:FF:FF:FF probe-req
 6   2412 -61dBm@1Mbps   00:01:24:70:3D:4E FF:FF:FF:FF:FF:FF probe-req
```

Snooper

Home menu level: ***/interface wireless snooper***

Description

With wireless snooper you can monitor the traffic load on each channel.

Property Description

channel-time (*time*; default: **200ms**) - how long to snoop each channel, if multiple-channels is set to yes

multiple-channels (yes | no; default: **no**) - whether to snoop multiple channels or a single channel

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **no** - wireless snooper snoops only one channel in frequency that is configured in /interface wireless
- **yes** - snoop in all channels that are listed in the scan-list in /interface wireless

receive-errors (yes | no; default: **no**) - whether to receive packets with CRC errors

Command Description

snoop - starts monitoring wireless channels

- **wireless interface name** - interface that monitoring is performed on
- **BAND** - operating band

Example

Snoop 802.11b network:

```
[admin@MikroTik] interface wireless snooper> snoop wlan1
BAND       FREQ    USE    BW         NET-COUNT STA-COUNT
2.4ghz-b   2412MHz 1.5%   11.8kbps   2         2
2.4ghz-b   2417MHz 1.3%   6.83kbps   0         1
2.4ghz-b   2422MHz 0.6%   4.38kbps   1         1
2.4ghz-b   2427MHz 0.6%   4.43kbps   0         0
2.4ghz-b   2432MHz 0.3%   2.22kbps   0         0
2.4ghz-b   2437MHz 0%     0bps       0         0
2.4ghz-b   2442MHz 1%     8.1kbps    0         0
2.4ghz-b   2447MHz 1%     8.22kbps   1         1
2.4ghz-b   2452MHz 1%     8.3kbps    0         0
2.4ghz-b   2457MHz 0%     0bps       0         0
2.4ghz-b   2462MHz 0%     0bps       0         0

[admin@MikroTik] interface wireless snooper>
```

General Information

Station and AccessPoint

This example shows how to configure 2 MikroTik routers - one as Access Point and the other one as a station on 5GHz (802.11a standard).

- On Access Point:
  - mode=ap-bridge
  - frequency=5805
  - band=5ghz
  - ssid=test
  - disabled=no

  On client (station):
  - mode=station
  - band=5ghz
  - ssid=test
  - disabled=no

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- Configure the Access Point and add an IP address (10.1.0.1) to it:

```
[admin@AccessPoint] interface wireless> set 0 mode=ap-bridge frequency=5805 \
   band=5ghz disabled=no ssid=test name=AP
[admin@AccessPoint] interface wireless> print
Flags: X - disabled, R - running
 0    name="AP" mtu=1500 mac-address=00:0C:42:05:00:22 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C42050022" mode=ap-bridge ssid="test" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=5805 band=5ghz scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@AccessPoint] interface wireless> /ip add
[admin@AccessPoint] ip address> add address=10.1.0.1/24 interface=AP
[admin@AccessPoint] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   10.1.0.1/24        10.1.0.0        10.1.0.255      AP
[admin@AccessPoint] ip address>
```

- Configure the station and add an IP address (10.1.0.2) to it:

```
[admin@Station] interface wireless> set wlan1 name=To-AP mode=station \
   ssid=test band=5ghz disabled=no
[admin@Station] interface wireless> print
Flags: X - disabled, R - running
 0 R  name="To-AP" mtu=1500 mac-address=00:0B:6B:34:5A:91 arp=enabled
      disable-running-check=no interface-type=Atheros AR5213
      radio-name="000B6B345A91" mode=station ssid="test" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=5180 band=5ghz scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@Station] interface wireless> /ip address
[admin@Station] ip address> add address=10.1.0.2/24 interface=To-AP
[admin@Station] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   172.16.0.2/24      172.16.0.0      172.16.0.255    To-AP
 1   192.168.2.3/24     192.168.2.0     192.168.2.255   To-AP
 2   10.1.0.2/24        10.1.0.0        10.1.0.255      To-AP
[admin@Station] ip address>
```

- Check whether you can ping the Access Point from Station:

```
[admin@Station] > ping 10.1.0.1
10.1.0.1 64 byte ping: ttl=64 time=3 ms
10.1.0.1 64 byte ping: ttl=64 time=3 ms
10.1.0.1 64 byte ping: ttl=64 time=3 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 3/3.0/3 ms
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@Station] >
```

WDS Station

Using 802.11 set of standards you cannot simply bridge wireless stations. To solve this problem, the **wds-station** mode was created - it works just like a station, but connects only to APs that support WDS.

This example shows you how to make a transparent network, using the Station WDS feature:

On WDS Access Point:

- Configure AP to support WDS connections
- Set **wds-default-bridge** to **bridge1**

On WDS station:

- Configure it as a WDS Station, using **mode=station-wds**

Configure the WDS Access Point. Configure the wireless interface and put it into a bridge, and define that the dynamic WDS links should be automatically put into the same bridge:

```
[admin@WDS_AP] > interface bridge
[admin@WDS_AP] interface bridge> add
[admin@WDS_AP] interface bridge> print
Flags: X - disabled, R - running
 0  R name="bridge1" mtu=1500 arp=enabled mac-address=B0:62:0D:08:FF:FF stp=no
      priority=32768 ageing-time=5m forward-delay=15s
      garbage-collection-interval=4s hello-time=2s max-message-age=20s
[admin@WDS_AP] interface bridge> port
[admin@WDS_AP] interface bridge port> print
 # INTERFACE BRIDGE  PRIORITY PATH-COST
 0 Public    none    128      10
 1 wlan1     none    128      10
[admin@WDS_AP] interface bridge port> set 0 bridge=bridge1
[admin@WDS_AP] interface bridge port> /in wireless
[admin@WDS_AP] interface wireless> set wlan1 mode=ap-bridge ssid=wds-sta-test \
   wds-mode=dynamic wds-default-bridge=bridge1 disabled=no band=2.4ghz-b/g \
   frequency=2437
[admin@WDS_AP] interface wireless> print
Flags: X - disabled, R - running
 0    name="wlan1" mtu=1500 mac-address=00:0C:42:05:00:22 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C42050022" mode=ap-bridge ssid="wds-sta-test" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=2437 band=2.4ghz-b/g scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=dynamic wds-default-bridge=bridge1 wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@WDS_AP] interface wireless>
```

Now configure the WDS station and put the wireless (**wlan1**) and ethernet (**Local**) interfaces into a bridge:

```
[admin@WDS_Station] > interface bridge
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@WDS_Station] interface bridge> add
[admin@WDS_Station] interface bridge> print
Flags: X - disabled, R - running
 0  R name="bridge1" mtu=1500 arp=enabled mac-address=11:05:00:00:02:00 stp=no
      priority=32768 ageing-time=5m forward-delay=15s
      garbage-collection-interval=4s hello-time=2s max-message-age=20s
[admin@WDS_Station] interface bridge> port
[admin@WDS_Station] interface bridge port> print
 # INTERFACE BRIDGE  PRIORITY PATH-COST
 0 Local     none    128      10
 1 wlan1     none    128      10
[admin@WDS_Station] interface bridge port> set 0,1 bridge=bridge1
[admin@WDS_Station] interface bridge port> /interface wireless
[admin@WDS_Station] interface wireless> set wlan1 mode=station-wds disabled=no \
\... ssid=wds-sta-test band=2.4ghz-b/g
[admin@WDS_Station] interface wireless> print
Flags: X - disabled, R - running
 0  R name="wlan1" mtu=1500 mac-address=00:0B:6B:34:5A:91 arp=enabled
      disable-running-check=no interface-type=Atheros AR5213
      radio-name="000B6B345A91" mode=station-wds ssid="wds-sta-test" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=2412 band=2.4ghz-b/g scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@WDS_Station] interface wireless>
```

Virtual Access Point

Virtual Access Point (VAP) enables you to create multiple Access Points with different Service Set Identifier, WDS settings, and even different MAC address, using the same hardware interface. You can create up to 7 VAP interfaces from a single physical interface. To create a Virtual Access Point, simply add a new interface, specifying a **master-interface** which is the physical interface that will do the hardware function to VAP.

This example will show you how to create a VAP:

```
[admin@VAP] interface wireless> print
Flags: X - disabled, R - running
 0    name="wlan1" mtu=1500 mac-address=00:0C:42:05:00:22 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C42050022" mode=ap-bridge ssid="test" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=2437 band=2.4ghz-b/g scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@VAP] interface wireless> add master-interface=wlan1 ssid=virtual-test \
\... mac-address=00:0C:42:12:34:56 disabled=no name=V-AP
[admin@VAP] interface wireless> print
Flags: X - disabled, R - running
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
 0    name="wlan1" mtu=1500 mac-address=00:0C:42:05:00:22 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C42050022" mode=ap-bridge ssid="test" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=2437 band=2.4ghz-b/g scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both

 1    name="V-AP" mtu=1500 mac-address=00:0C:42:12:34:56 arp=enabled
      disable-running-check=no interface-type=virtual-AP
      master-interface=wlan1 ssid="virtual-test" area=""
      max-station-count=2007 wds-mode=disabled wds-default-bridge=none
      wds-ignore-ssid=no default-authentication=yes default-forwarding=yes
      default-ap-tx-limit=0 default-client-tx-limit=0 hide-ssid=no
      security-profile=default
[admin@VAP] interface wireless>
```

When scanning from another router for an AP, you will see that you have 2 Access Points instead of one:

```
[admin@MikroTik] interface wireless> scan Station
Flags: A - active, B - bss, P - privacy, R - routeros-network, N - nstreme
      ADDRESS           SSID              BAND       FREQ SIG RADIO-NAME
AB R  00:0C:42:12:34:56 virtual-test      2.4ghz-g   2437 -72 000C42050022
AB R  00:0C:42:05:00:22 test              2.4ghz-g   2437 -72 000C42050022
-- [Q quit|D dump|C-z pause]
[admin@MikroTik] interface wireless>
```

Note that the **master-interface** must be configured as an Access Point (**ap-bridge** or **bridge** mode)!

Nstreme

This example shows you how to configure a point-to-point Nstreme link.

The setup of Nstreme is similar to usual wireless configuration, except that you have to do some changes under */interface wireless nstreme*.

- Set the **Nstreme-AP** to **bridge** mode and enable Nstreme on it:

```
[admin@Nstreme-AP] interface wireless> set 0 mode=bridge ssid=nstreme \
\... band=5ghz frequency=5805 disabled=no
[admin@Nstreme-AP] interface wireless> print
Flags: X - disabled, R - running
 0    name="wlan1" mtu=1500 mac-address=00:0C:42:05:00:22 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C42050022" mode=bridge ssid="nstreme" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=5805 band=5ghz scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@Nstreme-AP] interface wireless> nstreme
[admin@Nstreme-AP] interface wireless nstreme> set wlan1 enable-nstreme=yes
[admin@Nstreme-AP] interface wireless nstreme> print
 0 name="wlan1" enable-nstreme=yes enable-polling=yes framer-policy=none
   framer-limit=3200
[admin@Nstreme-AP] interface wireless nstreme>
```

- Configure **Nstreme-Client** wireless settings and enable Nstreme on it:

```
[admin@Nstreme-Client] interface wireless> set wlan1 mode=station ssid=nstreme \
   band=5ghz frequency=5805 disabled=no
[admin@Nstreme-Client] interface wireless> print
Flags: X - disabled, R - running
 0    name="wlan1" mtu=1500 mac-address=00:0B:6B:34:5A:91 arp=enabled
      disable-running-check=no interface-type=Atheros AR5213
      radio-name="000B6B345A91" mode=station ssid="nstreme" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=5805 band=5ghz scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@Nstreme-Client] interface wireless> nstreme
[admin@Nstreme-Client] interface wireless nstreme> set wlan1 enable-nstreme=yes
[admin@Nstreme-Client] interface wireless nstreme> print
 0 name="wlan1" enable-nstreme=yes enable-polling=yes framer-policy=none
   framer-limit=3200
[admin@Nstreme-Client] interface wireless nstreme>
```

And monitor the link:

```
[admin@Nstreme-Client] interface wireless> monitor wlan1
              status: connected-to-ess
                band: 5ghz
           frequency: 5805MHz
             tx-rate: 24Mbps
             rx-rate: 18Mbps
                ssid: "nstreme"
               bssid: 00:0C:42:05:00:22
          radio-name: "000C42050022"
     signal-strength: -70dBm
  tx-signal-strength: -68dBm
             tx-ccq: 0%
             rx-ccq: 3%
           wds-link: no
            nstreme: yes
            polling: yes
       framing-mode: none
    routeros-version: "2.9rc2"
   current-tx-powers: 1Mbps:11,2Mbps:11,5.5Mbps:11,11Mbps:11,6Mbps:28,
                      9Mbps:28,12Mbps:28,18Mbps:28,24Mbps:28,36Mbps:25,
                      48Mbps:23,54Mbps:22
-- [Q quit|D dump|C-z pause]
[admin@Nstreme-Client] interface wireless>
```

Dual Nstreme

The purpose of Nstreme2 (Dual Nstreme) is to make superfast point-to-point links, using 2 wireless cards on each router - one for receiving and the other one for transmitting data (you can use different bands for receiving and transmitting). This example will show you how to make a

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

point-to-point link, using Dual Nstreme.

Configure **DualNS-1**:

```
[admin@DualNS-1] interface wireless> set 0,1 mode=nstreme-dual-slave
[admin@DualNS-1] interface wireless> print
Flags: X - disabled, R - running
 0    name="wlan1" mtu=1500 mac-address=00:0C:42:05:04:36 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C42050436" mode=nstreme-dual-slave ssid="MikroTik"
      area="" frequency-mode=superchannel country=no_country_set
      antenna-gain=0 frequency=5180 band=5ghz scan-list=default
      rate-set=default supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both

 1    name="wlan2" mtu=1500 mac-address=00:0C:42:05:00:28 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C42050028" mode=nstreme-dual-slave ssid="MikroTik"
      area="" frequency-mode=superchannel country=no_country_set
      antenna-gain=0 frequency=5180 band=5ghz scan-list=default
      rate-set=default supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both

[admin@DualNS-1] interface wireless> nstreme-dual
[admin@DualNS-1] interface wireless nstreme-dual> add rx-radio=wlan1 \
   tx-radio=wlan2 rx-frequency=5180 tx-frequency=5805 disabled=no
[admin@DualNS-1] interface wireless nstreme-dual> print
Flags: X - disabled, R - running
 0  R name="nstreme1" mtu=1500 mac-address=00:0C:42:05:04:36 arp=enabled
      disable-running-check=no tx-radio=wlan2 rx-radio=wlan1
      remote-mac=00:00:00:00:00:00 tx-band=5ghz tx-frequency=5805
      rx-band=5ghz rx-frequency=5180 rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,54Mbps
      framer-policy=none framer-limit=4000
[admin@DualNS-1] interface wireless nstreme-dual>
```

Note the MAC address of the interface **nstreme1**. You will need it to configure the remote (**DualNS-2**) router. As we have not configured the **DualNS-2** router, we cannot define the **remote-mac** parameter on **DualNS-1**. We will do it after configuring **DualNS-2**!

The configuration of **DualNS-2**:

```
[admin@DualNS-2] interface wireless> set 0,1 mode=nstreme-dual-slave
[admin@DualNS-2] interface wireless> print
Flags: X - disabled, R - running
 0    name="wlan1" mtu=1500 mac-address=00:0C:42:05:00:22 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C42050022" mode=nstreme-dual-slave ssid="MikroTik"
      area="" frequency-mode=superchannel country=no_country_set
      antenna-gain=0 frequency=5180 band=5ghz scan-list=default
      rate-set=default supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both

 1    name="wlan2" mtu=1500 mac-address=00:0C:42:05:06:B2 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C420506B2" mode=nstreme-dual-slave ssid="MikroTik"
      area="" frequency-mode=superchannel country=no_country_set
      antenna-gain=0 frequency=5180 band=5ghz scan-list=default
      rate-set=default supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both

[admin@DualNS-2] interface wireless> nstreme-dual
[admin@DualNS-2] interface wireless nstreme-dual> add rx-radio=wlan1 \
\... tx-radio=wlan2 rx-frequency=5805 tx-frequency=5180 disabled=no \
\... remote-mac=00:0C:42:05:04:36
[admin@DualNS-2] interface wireless nstreme-dual> print
Flags: X - disabled, R - running
 0  R name="nstreme1" mtu=1500 mac-address=00:0C:42:05:00:22 arp=enabled
      disable-running-check=no tx-radio=wlan2 rx-radio=wlan1
      remote-mac=00:0C:42:05:04:36 tx-band=5ghz tx-frequency=5180
      rx-band=5ghz rx-frequency=5805 rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,54Mbps
      framer-policy=none framer-limit=4000
[admin@DualNS-2] interface wireless nstreme-dual>
```

Now complete the configuration for **DualNS-1**:

```
[admin@DualNS-1] interface wireless nstreme-dual> set 0 remote-mac=00:0C:42:05:00:22
[admin@DualNS-1] interface wireless nstreme-dual> print
Flags: X - disabled, R - running
 0  R name="nstreme1" mtu=1500 mac-address=00:0C:42:05:04:36 arp=enabled
      disable-running-check=no tx-radio=wlan2 rx-radio=wlan1
      remote-mac=00:0C:42:05:00:22 tx-band=5ghz tx-frequency=5805
      rx-band=5ghz rx-frequency=5180 rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,54Mbps
      framer-policy=none framer-limit=4000
[admin@DualNS-1] interface wireless nstreme-dual>
```

WEP Security

This example shows how to configure WEP (Wired Equivalent Privacy) on Access Point and Clients. In example we will configure an Access Point which will use **104bit-wep** for one station and **40bit-wep** for other clients. The configuration of stations is also present.

The key, used for connection between **WEP_AP** and **WEP_Station1** will be **6543210987654321098765432l**, key for **WEP_AP** and **WEP_StationX** will be **1234567890**!

Configure the Access Point:

```
[admin@WEP_AP] interface wireless security-profiles> add \
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
\... name=Station1 mode=static-keys-required static-sta-private-algo=104bit-wep \
\... static-sta-private-key=65432109876543210987654321
[admin@WEP_AP] interface wireless security-profiles> add name=StationX \
\... mode=static-keys-required static-algo-1=40bit-wep static-key-1=1234567890 \
\... static-transmit-key=key-1
[admin@WEP_AP] interface wireless security-profiles> print
 0 name="default" mode=none wpa-unicast-ciphers="" wpa-group-ciphers=""
   pre-shared-key="" static-algo-0=none static-key-0="" static-algo-1=none
   static-key-1="" static-algo-2=none static-key-2="" static-algo-3=none
   static-key-3="" static-transmit-key=key-0 static-sta-private-algo=none
   static-sta-private-key="" radius-mac-authentication=no group-key-update=5m

 1 name="Station1" mode=static-keys-required wpa-unicast-ciphers=""
   wpa-group-ciphers="" pre-shared-key="" static-algo-0=none static-key-0=""
   static-algo-1=none static-key-1="" static-algo-2=none static-key-2=""
   static-algo-3=none static-key-3="" static-transmit-key=key-0
   static-sta-private-algo=104bit-wep
   static-sta-private-key="65432109876543210987654321"
   radius-mac-authentication=no group-key-update=5m

 2 name="StationX" mode=static-keys-required wpa-unicast-ciphers=""
   wpa-group-ciphers="" pre-shared-key="" static-algo-0=none static-key-0=""
   static-algo-1=40bit-wep static-key-1="1234567890" static-algo-2=none
   static-key-2="" static-algo-3=none static-key-3=""
   static-transmit-key=key-1 static-sta-private-algo=none
   static-sta-private-key="" radius-mac-authentication=no group-key-update=5m
[admin@WEP_AP] interface wireless security-profiles> ..
[admin@MikroTik] interface wireless> set 0 name=WEP-AP mode=ap-bridge \
\... ssid=mt_wep frequency=5320 band=5ghz disabled=no security-profile=StationX
[admin@WEP_AP] interface wireless> print
Flags: X - disabled, R - running
 0    name="WEP-AP" mtu=1500 mac-address=00:0C:42:05:04:36 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C42050436" mode=ap-bridge ssid="mt_wep" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=5320 band=5ghz scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=StationX disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@WEP_AP] interface wireless> access-list
[admin@WEP_AP] interface wireless access-list> add private-algo=104bit-wep \
\... private-key=65432109876543210987654321 interface=WEP-AP forwarding=yes \
\... mac-address=00:0C:42:05:00:22
[admin@WEP_AP] interface wireless access-list> print
Flags: X - disabled
 0   mac-address=00:0C:42:05:00:22 interface=WEP-AP authentication=yes
     forwarding=yes ap-tx-limit=0 client-tx-limit=0 private-algo=104bit-wep
     private-key="65432109876543210987654321"
[admin@WEP_AP] interface wireless access-list>
```

Configure **WEP_StationX**:

```
[admin@WEP_Station1] interface wireless security-profiles> add name=Station1 \
\... mode=static-keys-required static-sta-private-algo=104bit-wep \
\... static-sta-private-key=65432109876543210987654321
[admin@WEP_Station1] interface wireless security-profiles> print
 0 name="default" mode=none wpa-unicast-ciphers="" wpa-group-ciphers=""
   pre-shared-key="" static-algo-0=none static-key-0="" static-algo-1=none
   static-key-1="" static-algo-2=none static-key-2="" static-algo-3=none
   static-key-3="" static-transmit-key=key-0 static-sta-private-algo=none
   static-sta-private-key="" radius-mac-authentication=no group-key-update=5m

 1 name="Station1" mode=static-keys-required wpa-unicast-ciphers=""
   wpa-group-ciphers="" pre-shared-key="" static-algo-0=none static-key-0=""
   static-algo-1=none static-key-1="" static-algo-2=none static-key-2=""
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
   static-algo-3=none static-key-3="" static-transmit-key=key-0
   static-sta-private-algo=104bit-wep
   static-sta-private-key="65432109876543210987654321"
   radius-mac-authentication=no group-key-update=5m
[admin@WEP_Station1] interface wireless security-profiles> ..
[admin@WEP_Station1] interface wireless> set wlan1 mode=station ssid=mt_wep \
\... band=5ghz security-profile=Station1 name=WEP-STA1 disabled=no
[admin@WEP_Station1] interface wireless> print
Flags: X - disabled, R - running
 0  R name="WEP-STA1" mtu=1500 mac-address=00:0C:42:05:00:22 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C42050022" mode=station ssid="mt_wep" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=5180 band=5ghz scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=Station1 disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@WEP_Station1] interface wireless>
```

Config of StationX:

```
[admin@WEP_StationX] interface wireless security-profiles> add name=StationX \
\... mode=static-keys-required static-algo-1=40bit-wep static-key-1=1234567890 \
\... static-transmit-key=key-1
[admin@WEP_StationX] interface wireless security-profiles> print
 0 name="default" mode=none wpa-unicast-ciphers="" wpa-group-ciphers=""
   pre-shared-key="" static-algo-0=none static-key-0="" static-algo-1=none
   static-key-1="" static-algo-2=none static-key-2="" static-algo-3=none
   static-key-3="" static-transmit-key=key-0 static-sta-private-algo=none
   static-sta-private-key="" radius-mac-authentication=no group-key-update=5m

 1 name="StationX" mode=static-keys-required wpa-unicast-ciphers=""
   wpa-group-ciphers="" pre-shared-key="" static-algo-0=none static-key-0=""
   static-algo-1=40bit-wep static-key-1="1234567890" static-algo-2=none
   static-key-2="" static-algo-3=none static-key-3=""
   static-transmit-key=key-1 static-sta-private-algo=none
   static-sta-private-key="" radius-mac-authentication=no group-key-update=5m
[admin@WEP_StationX] interface wireless security-profiles> ..
[admin@WEP_StationX] interface wireless> set wlan1 name=WEP-STAX ssid=mt_wep \
\... band=5ghz security-profile=StationX mode=station disabled=no
[admin@WEP_StationX] interface wireless> print
 0  R name="WEP-STAX" mtu=1500 mac-address=00:0C:42:05:06:B2 arp=enabled
      disable-running-check=no interface-type=Atheros AR5413
      radio-name="000C420506B2" mode=station ssid="mt_wep" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=5180 band=5ghz scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=StationX disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@WEP_StationX] interface wireless>
```

Troubleshooting

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

- **If I use WDS and DFS, the routers do not connect to each other!**
  As the WDS routers must operate at the same frequency, it is very probable that DFS will not select the frequency that is used by the peer router.
- **MikroTik RouterOS does not send any traffic through Cisco Wireless Access Point or Wireless Bridge**
  If you use CISCO/Aironet Wireless Ethernet Bridge or Access Point, you should set the Configuration/Radio/I80211/Extended (Allow proprietary extensions) to **off**, and the Configuration/Radio/I80211/Extended/Encapsulation (Default encapsulation method) to **RFC1042**. If left to the default **on** and **802.1H**, respectively, you won't be able to pass traffic through the bridge.
- **Prism wireless clients don't connect to AP after upgrade to 2.9**
  Prism wireless card's primary firmware version has to be at least 1.0.7 in order to boot wireless card's secondary firmware, which allows Prism card correctly operate under RouterOS. Check the log file to see whether the wireless card's secondary firmware was booted.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Xpeed SDSL Interface

Document revision 1.1 (Fri Mar 05 08:18:04 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Additional Documents
Xpeed Interface Configuration
  Property Description
  Example
Frame Relay Configuration Examples
  MikroTik Router to MikroTik Router
  MikroTik Router to Cisco Router
Troubleshooting
  Description

General Information

Summary

The MikroTik RouterOS supports the Xpeed 300 SDSL PCI Adapter hardware with speeds up to 2.32Mbps. This device can operate either using Frame Relay or PPP type of connection. SDSL (Single-line Digital Subscriber Line or Symmetric Digital Subscriber Line) stands for the type of DSL that uses only one of the two cable pairs for transmission. SDSL allows residential or small office users to share the same telephone for data transmission and voice or fax telephony.

Specifications

Packages required: ***synchronous***
License required: ***level4***
Home menu level: ***/interface xpeed***
Standards and Technologies: ***PPP (RFC 1661)***, ***Frame Relay (RFC 1490)***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***
- ***Xpeed SDSL Interface***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Additional Documents

- ***Xpeed homepage***

Xpeed Interface Configuration

Home menu level: ***/interface xpeed***

Property Description

name (*name*) - interface name

mtu (*integer*; default: **1500**) - Maximum Transmission Unit

mac-address (*MAC address*) - MAC address of the card

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol

- **disabled** - the interface will not use ARP protocol
- **enabled** - the interface will use ARP protocol
- **proxy-arp** - the interface will be an ARP proxy
- **reply-only** - the interface will only reply to the requests originated to its own IP addresses, but neighbor MAC addresses will be gathered from /ip arp statically set table only

mode (*network-termination* | *line-termination*; default: **line-termination**) - interface mode, either line termination (LT) or network termination (NT)

sdsl-speed (*integer*; default: **2320**) - SDSL connection speed

sdsl-invert (*yes* | *no*; default: **no**) - whether the clock is phase inverted with respect to the Transmitted Data interchange circuit. This configuration option is useful when long cable lengths between the Termination Unit and the DTE are causing data errors

sdsl-swap (*yes* | *no*; default: **no**) - whether or not the Xpeed 300 SDSL Adapter performs bit swapping. Bit swapping can maximize error performance by attempting to maintain an acceptable margin for each bin by equalizing the margin across all bins through bit reallocation

bridged-ethernet (*yes* | *no*; default: **yes**) - if the adapter operates in bridged Ethernet mode

dlci (*integer*; default: **16**) - defines the DLCI to be used for the local interface. The DLCI field identifies which logical circuit the data travels over

lmi-mode (*off* | *line-termination* | *network-termination* | *network-termination-bidirectional*; default: **off**) - defines how the card will perform LMI protocol negotiation

- **off** - no LMI will be used
- **line-termination** - LMI will operate in LT (Line Termination) mode
- **network-termination** - LMI will operate in NT (Network Termination) mode
- **network-termination-bidirectional** - LMI will operate in bidirectional NT mode

cr (*0* | *2*; default: **0**) - a special mask value to be used when speaking with certain buggy vendor equipment. Can be 0 or 2

Example

To enable interface:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@r1] interface> print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                                                TYPE            MTU
  0  R outer                                               ether           1500
  1  R inner                                               ether           1500
  2 X  xpeed1                                              xpeed           1500

[admin@r1] interface> enable 2
[admin@r1] interface> print
Flags: X - disabled, D - dynamic, R - running
  #    NAME                                                TYPE            MTU
  0  R outer                                               ether           1500
  1  R inner                                               ether           1500
  2  R xpeed1                                              xpeed           1500

[admin@r1] interface>
```

Frame Relay Configuration Examples

MikroTik Router to MikroTik Router

Consider the following network setup with MikroTik router connected via SDSL line using Xpeed interface to another MikroTik router with Xpeed 300 SDSL adapter. SDSL line can refer a common patch cable included with the Xpeed 300 SDSL adapter (such a connection is called Back-to-Back). Lets name the first router **r1** and the second **r2**.

Router **r1** setup

The following setup is identical to one in the first example:

```
[admin@r1] ip address> add inter=xpeed1 address 1.1.1.1/24
[admin@r1] ip address> pri
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   1.1.1.1/24         1.1.1.0         1.1.1.255       xpeed1

[admin@r1] interface xpeed> print
Flags: X - disabled
  0   name="xpeed1" mtu=1500 mac-address=00:05:7A:00:00:08 arp=enabled
      mode=network-termination sdsl-speed=2320 sdsl-invert=no sdsl-swap=no
      bridged-ethernet=yes dlci=16 lmi-mode=off cr=0
[admin@r1] interface xpeed>
```

Router **r2** setup

First, we need to add a suitable IP address:

```
[admin@r2] ip address> add inter=xpeed1 address 1.1.1.2/24
[admin@r2] ip address> pri
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   1.1.1.2/24         1.1.1.0         1.1.1.255       xpeed1
```

Then, some changes in **xpeed** interface configuration should be done:

```
[admin@r2] interface xpeed> print
Flags: X - disabled
  0   name="xpeed1" mtu=1500 mac-address=00:05:7A:00:00:08 arp=enabled
      mode=network-termination sdsl-speed=2320 sdsl-invert=no sdsl-swap=no
      bridged-ethernet=yes dlci=16 lmi-mode=off cr=0
[admin@r2] interface xpeed> set 0 mode=line-termination
[admin@r2] interface xpeed>
```

Now **r1** and **r2** can ping each other.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

MikroTik Router to Cisco Router

Let us consider the following network setup with MikroTik Router with Xpeed interface connected to a leased line with a CISCO router at the other end.

MikroTik router setup:

```
[admin@r1] ip address> add inter=xpeed1 address 1.1.1.1/24
[admin@r1] ip address> pri
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   1.1.1.1/24         1.1.1.0         1.1.1.255       xpeed1

[admin@r1] interface xpeed> print
Flags: X - disabled
  0   name="xpeed1" mtu=1500 mac-address=00:05:7A:00:00:08 arp=enabled
      mode=network-termination sdsl-speed=2320 sdsl-invert=no sdsl-swap=no
      bridged-ethernet=yes dlci=42 lmi-mode=off cr=0
[admin@r1] interface xpeed>
```

Cisco router setup

```
CISCO# show running-config
Building configuration...
Current configuration...

...
!
ip subnet-zero
no ip domain-lookup
frame-relay switching
!
interface Ethernet0
 description connected to EthernetLAN
 ip address 10.0.0.254 255.255.255.0
!
interface Serial0
 description connected to Internet
 no ip address
 encapsulation frame-relay IETF
 serial restart-delay 1
 frame-relay lmi-type ansi
 frame-relay intf-type dce
!
interface Serial0.1 point-to-point
 ip address 1.1.1.2 255.255.255.0
 no arp frame-relay
 frame-relay interface-dlci 42
!
...
end.

Send ping to MikroTik router

CISCO#ping 1.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/31/32 ms
CISCO#
```

Troubleshooting

Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **I tried to connect two routers as shown in MT-to-MT, but nothing happens**
  The link indicators on both cards must be on. If it's not, check the cable or interface configuration. One adapter should use LT mode and the other NT mode. You can also change **sdsl-swap** and **sdsl-invert** parameters on the router running LT mode if you have a very long line

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

EoIP Tunnel Interface

Document revision 1.3 (Tue Mar 09 08:15:37 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Quick Setup Guide
  Specifications
  Related Documents
  Description
EoIP Setup
  Property Description
  Notes
  Example
EoIP Application Example
  Description
  Example
Troubleshooting
  Description

General Information

Summary

Ethernet over IP (EoIP) Tunneling is a MikroTik RouterOS protocol that creates an Ethernet tunnel between two routers on top of an IP connection. The EoIP interface appears as an Ethernet interface. When the bridging function of the router is enabled, all Ethernet traffic (all Ethernet protocols) will be bridged just as if there where a physical Ethernet interface and cable between the two routers (with bridging enabled). This protocol makes multiple network schemes possible.

Network setups with EoIP interfaces:

- Possibility to bridge LANs over the Internet
- Possibility to bridge LANs over encrypted tunnels
- Possibility to bridge LANs over 802.11b 'ad-hoc' wireless networks

Quick Setup Guide

To make an EoIP tunnel between 2 routers which have IP addresses **10.5.8.1** and **10.1.0.1**:

1. On router with IP address **10.5.8.1**, add an EoIP interface and set its MAC address:

```
/interface eoip add remote-address=10.1.0.1 tunnel-id=1 mac-address=00-00-5E-80-00-01 \
\... disabled=no
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

2. On router with IP address **10.1.0.1**, add an EoIP interface and set its MAC address::

```
/interface eoip add remote-address=10.5.8.1 tunnel-id=1 mac-address=00-00-5E-80-00-02 \
\... disabled=no
```

Now you can add IP addresses to the created EoIP interfaces from the same subnet.

Specifications

Packages required: ***system***
License required: ***level1 (limited to 1 tunnel), level3***
Home menu level: ***/interface eoip***
Standards and Technologies: ***GRE (RFC1701)***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***IP Addresses and ARP***
- ***Bridge Interfaces***
- ***PPTP Interface***

Description

An EoIP interface should be configured on two routers that have the possibility for an IP level connection. The EoIP tunnel may run over an IPIP tunnel, a PPTP 128bit encrypted tunnel, a PPPoE connection, or any connection that transports IP.

Specific Properties:

- Each EoIP tunnel interface can connect with one remote router which has a corresponding interface configured with the same 'Tunnel ID'.
- The EoIP interface appears as an Ethernet interface under the interface list.
- This interface supports all features of an Ethernet interface. IP addresses and other tunnels may be run over the interface.
- The EoIP protocol encapsulates Ethernet frames in GRE (IP protocol number 47) packets (just like PPTP) and sends them to the remote side of the EoIP tunnel.
- Maximal count of EoIP tunnels is 65536.

EoIP Setup

Home menu level: ***/interface eoip***

Property Description

name (*name*; default: **eoip-tunnelN**) - interface name for reference

mtu (*integer*; default: **1500**) - Maximum Transmission Unit. The default value provides maximal compatibility

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol

tunnel-id (*integer*) - a unique tunnel identifier

remote-address - the IP address of the other side of the EoIP tunnel - must be a MikroTik router

mac-address (*MAC address*) - MAC address of the EoIP interface. You can freely use MAC addresses that are in the range from 00-00-5E-80-00-00 to 00-00-5E-FF-FF-FF

Notes

tunnel-id is method of identifying tunnel. There should not be tunnels with the same **tunnel-id** on the same router. **tunnel-id** on both participant routers must be equal.

mtu should be set to 1500 to eliminate packet refragmentation inside the tunnel (that allows transparent bridging of Ethernet-like networks, so that it would be possible to transport full-sized Ethernet frame over the tunnel).

For **EoIP** interfaces you can use MAC addresses that are in the range from **00-00-5E-80-00-00** to **00-00-5E-FF-FF-FF**.

Example

To add and enable an EoIP tunnel named **to_mt2** to the **10.5.8.1** router, specifying **tunnel-id** of **1**:

```
[admin@MikroTik] interface eoip> add name=to_mt2 remote-address=10.5.8.1 \
\... tunnel-id 1
[admin@MikroTik] interface eoip> print
Flags: X - disabled, R - running
  0 X  name="to_mt2" mtu=1500 arp=enabled remote-address=10.5.8.1 tunnel-id=1

[admin@MikroTik] interface eoip> enable 0
[admin@MikroTik] interface eoip> print
Flags: X - disabled, R - running
  0 R  name="to_mt2" mtu=1500 arp=enabled remote-address=10.5.8.1 tunnel-id=1

[admin@MikroTik] interface eoip>
```

EoIP Application Example

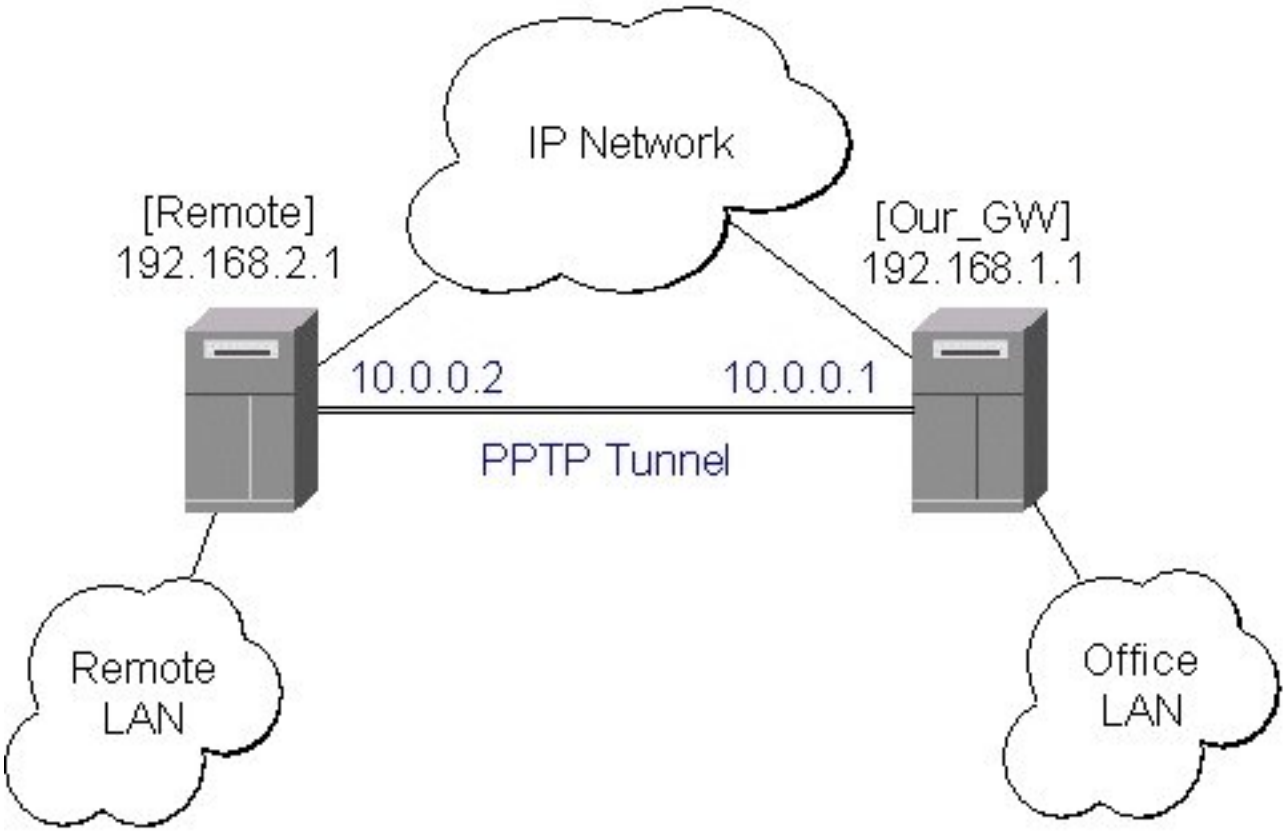
Description

Let us assume we want to bridge two networks: 'Office LAN' and 'Remote LAN'. The networks are connected to an IP network through the routers [Our_GW] and [Remote]. The IP network can be a private intranet or the Internet. Both routers can communicate with each other through the IP network.

Example

Our goal is to create a secure channel between the routers and bridge both networks through it. The network setup diagram is as follows:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



To make a secure Ethernet bridge between two routers you should:

1. Create a PPTP tunnel between them. Our_GW will be the pptp server:

```
[admin@Our_GW] interface pptp-server> /ppp secret add name=joe service=pptp \
\... password=top_s3 local-address=10.0.0.1 remote-address=10.0.0.2
[admin@Our_GW] interface pptp-server> add name=from_remote user=joe
[admin@Our_GW] interface pptp-server> server set enable=yes
[admin@Our_GW] interface pptp-server> print
Flags: X - disabled, D - dynamic, R - running
  #     NAME                USER         MTU   CLIENT-ADDRESS  UPTIME   ENC...
  0     from_remote         joe
[admin@Our_GW] interface pptp-server>

The Remote router will be the pptp client:

[admin@Remote] interface pptp-client> add name=pptp user=joe \
\... connect-to=192.168.1.1 password=top_s3 mtu=1500 mru=1500
[admin@Remote] interface pptp-client> enable pptp
[admin@Remote] interface pptp-client> print
Flags: X - disabled, R - running
  0  R name="pptp" mtu=1500 mru=1500 connect-to=192.168.1.1 user="joe"
       password="top_s2" profile=default add-default-route=no

[admin@Remote] interface pptp-client> monitor pptp
      status: "connected"
      uptime: 39m46s
    encoding: "none"

[admin@Remote] interface pptp-client>
```

   See the PPTP Interface Manual for more details on setting up encrypted channels.

2. Configure the EoIP tunnel by adding the eoip tunnel interfaces at both routers. Use the ip addresses of the pptp tunnel interfaces when specifying the argument values for the EoIP

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

tunnel:

```
[admin@Our_GW] interface eoip> add name="eoip-remote" tunnel-id=0 \
\... remote-address=10.0.0.2
[admin@Our_GW] interface eoip> enable eoip-remote
[admin@Our_GW] interface eoip> print
Flags: X - disabled, R - running
  0    name=eoip-remote mtu=1500 arp=enabled remote-address=10.0.0.2 tunnel-id=0
[admin@Our_GW] interface eoip>

[admin@Remote] interface eoip> add name="eoip" tunnel-id=0 \
\... remote-address=10.0.0.1
[admin@Remote] interface eoip> enable eoip-main
[admin@Remote] interface eoip> print
Flags: X - disabled, R - running
  0   name=eoip mtu=1500 arp=enabled remote-address=10.0.0.1 tunnel-id=0

[Remote] interface eoip>
```

3. Enable bridging between the EoIP and Ethernet interfaces on both routers.
   On the Our_GW:

```
[admin@Our_GW] interface bridge> add forward-protocols=ip,arp,other \
\... disabled=no
[admin@Our_GW] interface bridge> print
Flags: X - disabled, R - running
  0  R name="bridge1" mtu=1500 arp=enabled mac-address=00:00:00:00:00:00
       forward-protocols=ip,arp,other priority=1

[admin@Our_GW] interface bridge> port print
Flags: X - disabled
  #   INTERFACE                           BRIDGE
  0   eoip-remote                         none
  1   office-eth                          none
  2   isp                                 none

[admin@Our_GW] interface bridge> port set "0,1" bridge=bridge1
```

   And the same for the Remote:

```
[admin@Remote] interface bridge> add forward-protocols=ip,arp,other \
\... disabled=no
[admin@Remote] interface bridge> print
Flags: X - disabled, R - running
  0  R name="bridge1" mtu=1500 arp=enabled mac-address=00:00:00:00:00:00
       forward-protocols=ip,arp,other priority=1

[admin@Remote] interface bridge> port print
Flags: X - disabled
  #   INTERFACE                           BRIDGE
  0   ether                               none
  1   adsl                                none
  2   eoip-main                           none

[admin@Remote] interface bridge> port set "0,2" bridge=bridge1
```

4. Addresses from the same network can be used both in the Office LAN and in the Remote LAN.

Troubleshooting

Description

- **The routers can ping each other but EoIP tunnel does not seem to work!**
  Check the MAC addresses of the EoIP interfaces - they should not be the same!

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

IP Security

Document revision 3.3 (Fri May 06 09:14:43 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Specifications
  Related Documents
  Description
Policy Settings
  Description
  Property Description
  Notes
  Example
Peers
  Description
  Property Description
  Notes
  Example
Remote Peer Statistics
  Description
  Property Description
  Example
Installed SAs
  Description
  Property Description
  Example
Flushing Installed SA Table
  Description
  Property Description
  Example
Counters
  Property Description
  Example
  MikroTik Router to MikroTik Router
  IPsec Between two Masquerading MikroTik Routers
  MikroTik router to CISCO Router
  MikroTik Router and Linux FreeS/WAN

General Information

Specifications

Packages required: ***security***
License required: ***level1***
Home menu level: ***/ip ipsec***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Standards and Technologies: ***IPsec***
Hardware usage: *consumes a lot of CPU time (Intel Pentium MMX or AMD K6 suggested as a minimal configuration)*

Related Documents

- ***Software Package Management***

Description

IPsec (IP Security) supports secure (encrypted) communications over IP networks.

Encryption

After packet is src-natted, but before putting it into interface queue, IPsec policy database is consulted to find out if packet should be encrypted. Security Policy Database (SPD) is a list of rules that have two parts:

- **Packet matching** - packet source/destination, protocol and ports (for TCP and UDP) are compared to values in policy rules, one after another
- **Action** - if rule matches action specified in rule is performed:
- **accept** - continue with packet as if there was no IPsec
- **drop** - drop packet
- **encrypt** - encrypt packet

Each SPD rule can be associated with several Security Associations (SA) that determine packet encryption parameters (key, algorithm, SPI).

Note that packet can only be encrypted if there is usable SA for policy rule. By setting SPD rule security "level" user can control what happens when there is no valid SA for policy rule:

- **use** - if there is no valid SA, send packet unencrypted (like accept rule)
- **acquire** - send packet unencrypted, but ask IKE daemon to establish new SA
- **require** - drop packet, and ask IKE daemon to establish new SA.

Decryption

When encrypted packet is received for local host (after **dst-nat** and **input** filter), the appropriate SA is looked up to decrypt it (using packet source, destination, security protocol and SPI value). If no SA is found, the packet is dropped. If SA is found, packet is decrypted. Then decrypted packet's fields are compared to policy rule that SA is linked to. If the packet does not match the policy rule it is dropped. If the packet is decrypted fine (or authenticated fine) it is "received once more" - it goes through **dst-nat** and routing (which finds out what to do - either forward or deliver locally) again.

Note that before **forward** and **input** firewall chains, a packet that was not decrypted on local host is compared with SPD reversing its matching rules. If SPD requires encryption (there is valid SA associated with matching SPD rule), the packet is dropped. This is called incoming policy check.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Internet Key Exchange

The Internet Key Exchange (IKE) is a protocol that provides authenticated keying material for Internet Security Association and Key Management Protocol (ISAKMP) framework. There are other key exchange schemes that work with ISAKMP, but IKE is the most widely used one. Together they provide means for authentication of hosts and automatic management of security associations (SA).

Most of the time IKE daemon is doing nothing. There are two possible situations when it is activated:

- There is some traffic caught by a policy rule which needs to become encrypted or authenticated, but the policy doesn't have any SAs. The policy notifies IKE daemon about that, and IKE daemon initiates connection to remote host.
- IKE daemon responds to remote connection.

In both cases, peers establish connection and execute 2 phases:

- **Phase 1** - The peers agree upon algorithms they will use in the following IKE messages and authenticate. The keying material used to derive keys for all SAs and to protect following ISAKMP exchanges between hosts is generated also.
- **Phase 2** - The peers establish one or more SAs that will be used by IPsec to encrypt data. All SAs established by IKE daemon will have lifetime values (either limiting time, after which SA will become invalid, or amount of data that can be encrypted by this SA, or both).

There are two lifetime values - soft and hard. When SA reaches it's soft lifetime treshold, the IKE daemon receives a notice and starts another phase 2 exchange to replace this SA with fresh one. If SA reaches hard lifetime, it is discarded.

IKE can optionally provide a Perfect Forward Secrecy (PFS), whish is a property of key exchanges, that, in turn, means for IKE that compromising the long term phase 1 key will not allow to easily gain access to all IPsec data that is protected by SAs established through this phase 1. It means an additional keying material is generated for each phase 2.

Generation of keying material is computationally very expensive. *Exempli gratia*, the use of modp8192 group can take several seconds even on very fast computer. It usually takes place once per phase 1 exchange, which happens only once between any host pair and then is kept for long time. PFS adds this expensive operation also to each phase 2 exchange.

Diffie-Hellman MODP Groups

Diffie-Hellman (DH) key exchange protocol allows two parties without any initial shared secret to create one securely. The following Modular Exponential (MODP) Diffie-Hellman (also known as "Oakley") Groups are supported:

Diffie-Hellman Group	Modulus	Reference
Group 1	**768 bits**	**RFC2409**
Group 2	**1024 bits**	**RFC2409**
Group 5	**1536 bits**	**RFC3526**

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

IKE Traffic

To avoid problems with IKE packets hit some SPD rule and require to encrypt it with not yet established SA (that this packet perhaps is trying to establish), locally originated packets with UDP source port 500 are not processed with SPD. The same way packets with UDP destination port 500 that are to be delivered locally are not processed in incoming policy check.

Setup Procedure

To get IPsec to work with automatic keying using IKE-ISAKMP you will have to configure **policy**, **peer** and **proposal** (optional) entries.

For manual keying you will have to configure **policy** and **manual-sa** entries.

Policy Settings

Home menu level: ***/ip ipsec policy***

Description

Policy table is needed to determine whether encryption should be applied to a packet.

Property Description

action (*accept* | *drop* | *encrypt*; default: **accept**) - specifies what action to undertake with a packet that matches the policy

- **accept** - pass the packet
- **drop** - drop the packet
- **encrypt** - apply transformations specified in this policy and it's SA

decrypted (*integer*) - how many incoming packets were decrypted by the policy

dont-fragment (*clear* | *inherit* | *set*; default: **clear**) - The state of the don't fragment IP header field

- **clear** - clear (unset) the fields, so that packets previously marked as don't fragment got fragmented
- **inherit** - do not change the field
- **set** - set the field, so that each packet matching the rule will not be fragmented

dst-address (*IP address* | *netmask* | *port*; default: **0.0.0.0/32:any**) - destination IP address

encrypted (*integer*) - how many outgoing packets were encrypted by the policy

in-accepted (*integer*) - how many incoming packets were passed through by the policy without an attempt to decrypt

in-dropped (*integer*) - how many incoming packets were dropped by the policy without an attempt to decrypt

ipsec-protocols (*multiple choice: ah* | *esp*; default: **esp**) - specifies what combination of Authentication Header and Encapsulating Security Payload protocols you want to apply to matched traffic. AH is applied after ESP, and in case of tunnel mode ESP will be applied in tunnel mode and AH - in transport mode

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

level (*acquire* | *require* | *use*; default: **require**) - specifies what to do if some of the SAs for this policy cannot be found:

- **use** - skip this transform, do not drop packet and do not acquire SA from IKE daemon
- **acquire** - skip this transform, but acquire SA for it from IKE daemon
- **require** - drop packet but acquire SA

manual-sa (*name*; default: **none**) - name of manual-sa template that will be used to create SAs for this policy

- **none** - no manual keys are set

not-decrypted (*integer*) - how many incoming packets the policy attempted to decrypt. but discarded for any reason

not-encrypted (*integer*) - how many outgoing packets the policy attempted to encrypt. but discarded for any reason

out-accepted (*integer*) - how many outgoing packets were passed through by the policy without an attempt to encrypt

out-dropped (*integer*) - how many outgoing packets were dropped by the policy without an attempt to encrypt

ph2-state (*read-only: expired* | *no-phase2* | *established*) - indication of the progress of key establishing

- **expired** - there are some leftovers from previous phase2. In general it is similar to no-phase2
- **no-phase2** - no keys are estabilished at the moment
- **estabilished** - Appropriate SAs are in place and everything should be working fine

proposal (*name*; default: **default**) - name of proposal information that will be sent by IKE daemon to establish SAs for this policy

protocol (*name* | *integer*; default: **all**) - protocol name or number

sa-dst-address (*IP address*; default: **0.0.0.0**) - SA destination IP address

sa-src-address (*IP address*; default: **0.0.0.0**) - SA source IP address

src-address (*IP address* | *netmask* | *port*; default: **0.0.0.0/32:any**) - source IP address

tunnel (yes | no; default: **no**) - specifies whether to use tunnel mode

Notes

All packets are IPIP encapsulated in tunnel mode, and their new IP header **src-address** and **dst-address** are set to **sa-src-address** and **sa-dst-address** values of this policy. If you do not use tunnel mode (*id est* you use transport mode), then only packets whose source and destination addresses are the same as **sa-src-address** and **sa-dst-address** can be processed by this policy. Transport mode can only work with packets that originate at and are destined for IPsec peers (hosts that established security associations). To encrypt traffic between networks (or a network and a host) you have to use tunnel mode.

It is good to have **dont-fragment** cleared because encrypted packets are always bigger than original and thus they may need fragmentation.

If you are using IKE to establish SAs automatically, then policies on both routers must exactly match each other, *id est* **src-address=1.2.3.0/27** on one router and **dst-address=1.2.3.0/28** on another would not work. Source address values on one router MUST be equal to destination address

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

values on the other one, and vice versa.

Example

To add a policy to encrypt all the traffic between two hosts (10.0.0.147 and 10.0.0.148), we need do the following:

```
[admin@WiFi] ip ipsec policy> add sa-src-address=10.0.0.147 \
\... sa-dst-address=10.0.0.148 action=encrypt
[admin@WiFi] ip ipsec policy> print
Flags: X - disabled, D - dynamic, I - invalid
 0   src-address=10.0.0.147/32:any dst-address=10.0.0.148/32:any protocol=all
     action=encrypt level=require ipsec-protocols=esp tunnel=no
     sa-src-address=10.0.0.147 sa-dst-address=10.0.0.148 proposal=default
     manual-sa=none dont-fragment=clear

[admin@WiFi] ip ipsec policy>
```

to view the policy statistics, do the following:

```
[admin@WiFi] ip ipsec policy> print stats
Flags: X - disabled, D - dynamic, I - invalid
  0   src-address=10.0.0.147/32:any dst-address=10.0.0.148/32:any
      protocol=all ph2-state=no-phase2 in-accepted=0 in-dropped=0
      out-accepted=0 out-dropped=0 encrypted=0 not-encrypted=0 decrypted=0
      not-decrypted=0

[admin@WiFi] ip ipsec policy>
```

Peers

Home menu level: ***/ip ipsec peer***

Description

Peer configuration settings are used to establish connections between IKE daemons (phase 1 configuration). This connection then will be used to negotiate keys and algorithms for SAs.

Property Description

address (*IP address* | *netmask* | *port*; default: **0.0.0.0/32:500**) - address prefix. If remote peer's address matches this prefix, then this peer configuration is used while authenticating and establishing phase 1. If several peer's addresses matches several configuration entries, the most specific one (i.e. the one with largest netmask) will be used

dh-group (*multiple choice: modp768* | *modp1024* | *modp1536*; default: **esp**) - Diffie-Hellman MODP group (cipher strength)

enc-algorithm (*multiple choice: des* | *3des* | *aes-128* | *aes-192* | *aes-256*; default: **3des**) - encryption algorithm. Algorithms are named in strength increasing order

exchange-mode (*multiple choice: main* | *aggressive* | *base*; default: **main**) - different ISAKMP phase 1 exchange modes according to RFC 2408.DO not use other modes then main unless you know what you are doing

generate-policy (yes | no; default: **no**) - allow this peer to establish SA for non-existing policies. Such policies are created dynamically for the lifetime of SA. This way it is possible, for example, to create IPsec secured L2TP tunnels, or any other setup where remote peer's IP address is not known at configuration time

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

hash-algorithm (*multiple choice: md5* | *sha*; default: **md5**) - hashing algorithm. SHA (Secure Hash Algorithm) is stronger, but slower

lifebytes (*integer*; default: **0**) - phase 1 lifetime: specifies how much bytes can be transferred before SA is discarded

- **0** - SA expiration will not be due to byte count excess

lifetime (*time*; default: **1d**) - phase 1 lifetime: specifies how long the SA will be valid; SA will be discarded after this time

proposal-check (*multiple choice: claim* | *exact* | *obey* | *strict*; default: **strict**) - phase 2 lifetime check logic:

- **claim** - take shortest of proposed and configured lifetimes and notify initiator about it
- **exact** - require lifetimes to be the same
- **obey** - accept whatever is sent by an initiator
- **strict** - If proposed lifetime IS longer than default then reject proposal otherwise accept proposed lifetime

secret (*text*; default: **""**) - secret string. If it starts with '0x', it is parsed as a hexadecimal value

send-initial-contact (yes | no; default: **yes**) - specifies whether to send initial IKE information or wait for remote side

Notes

AES (Advanced Encryption Standard) encryption algorithms are much faster than DES, so it is recommended to use this algorithm class whenever possible. But, AES's speed is also its drawback as it potentially can be cracked faster, so use AES-256 when you need security or AES-128 when speed is also important.

Both peers MUST have the same encryption and authentication algorithms, DH group and exchange mode. Some legacy hardware may support only DES and MD5.

You should set **generate-policy** flag to **yes** only for trusted peers, because there is no verification done for the established policy. To protect yourself against possible unwanted events, add policies with **action=accept** for all networks you don't want to be encrypted at the top of policy list. Since dynamic policies are added at the bottom of the list, they will not be able to override your configuration.

Example

To define new peer configuration for **10.0.0.147** peer with **secret=gwejimezyfopmekun**:

```
[admin@WiFi] ip ipsec peer>add address=10.0.0.147/32 \
\... secret=gwejimezyfopmekun
[admin@WiFi] ip ipsec peer> print
Flags: X - disabled
  0   address=10.0.0.147/32:500 secret="gwejimezyfopmekun" generate-policy=no
      exchange-mode=main send-initial-contact=yes proposal-check=obey
      hash-algorithm=md5 enc-algorithm=3des dh-group=modp1024 lifetime=1d
      lifebytes=0

[admin@WiFi] ip ipsec peer>
```

Remote Peer Statistics

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Home menu level: ***/ip ipsec remote-peers***

Description

This submenu provides you with various statistics about remote peers that currently have established phase 1 connections with this router. Note that if peer doesn't show up here, it doesn't mean that no IPsec traffic is being exchanged with it. For example, manually configured SAs will not show up here.

Property Description

estabilished (*read-only: text*) - shows date and time when phase 1 was established with the peer

local-address (*read-only: IP address*) - local ISAKMP SA address

ph2-active (*read-only: integer*) - how many phase 2 negotiations with this peer are currently taking place

ph2-total (*read-only: integer*) - how many phase 2 negotiations with this peer took place

remote-address (*read-only: IP address*) - peer's IP address

side (*multiple choice, read-only: initiator | responder*) - shows which side initiated the connection

- **initiator** - phase 1 negotiation was started by this router
- **responder** - phase 1 negotiation was started by peer

state (*read-only: text*) - state of phase 1 negotiation with the peer

- **estabilished** - normal working state

Example

To see currently estabilished SAs:

```
[admin@WiFi] ip ipsec> remote-peers print
  0 local-address=10.0.0.148 remote-address=10.0.0.147 state=established
    side=initiator established=jan/25/2003 03:34:45 ph2-active=0 ph2-total=1
[admin@WiFi] ip ipsec>
```

Installed SAs

Home menu level: ***/ip ipsec installed-sa***

Description

This facility provides information about installed security associations including the keys

Property Description

add-lifetime (*read-only: time*) - soft/hard expiration time counted from installation of SA

auth-algorithm (*multiple choice, read-only: none | md5 | sha1*) - authentication algorithm used in SA

auth-key (*read-only: text*) - authentication key presented in form of hex string

current-addtime (*read-only: text*) - time when this SA was installed

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

current-bytes (*read-only: integer*) - amount of data processed by this SA's crypto algorithms

current-usetime (*read-only: text*) - time when this SA was first used

direction (*multiple choice, read-only: in | out*) - SA direction

dst-address (*read-only: IP address*) - destination address of SA taken from respective policy

enc-algorithm (*multiple choice, read-only: none | des | 3des | aes*) - encryption algorithm used in SA

enc-key (*read-only: text*) - encryption key presented in form of hex string (not applicable to AH SAs)

lifebytes (*read-only: integer*) - soft/hard expiration threshold for amount of processed data

replay (*read-only: integer*) - size of replay window presented in bytes. This window protects the receiver against replay attacks by rejecting old or duplicate packets.

spi (*read-only: integer*) - SPI value of SA, represented in hexadecimal form

src-address (*read-only: IP address*) - source address of SA taken from respective policy

state (*multiple choice, read-only: larval | mature | dying | dead*) - SA living phase

use-lifetime (*read-only: time*) - soft/hard expiration time counted from the first use of SA

Example

Sample printout looks as follows:

```
[admin@WiFi] ip ipsec> installed-sa print
Flags: A - AH, E - ESP, P - pfs, M - manual
  0 E   spi=E727605 direction=in src-address=10.0.0.148
        dst-address=10.0.0.147 auth-algorithm=sha1 enc-algorithm=3des
        replay=4 state=mature
        auth-key="ecc5f4aee1b297739ec88e324d7cfb8594aa6c35"
        enc-key="d6943b8ea582582e449bde085c9471ab0b209783c9eb4bbd"
        add-lifetime=24m/30m use-lifetime=0s/0s lifebytes=0/0
        current-addtime=jan/28/2003 20:55:12
        current-usetime=jan/28/2003 20:55:23 current-bytes=128
  1 E   spi=E15CEE06 direction=out src-address=10.0.0.147
        dst-address=10.0.0.148 auth-algorithm=sha1 enc-algorithm=3des
        replay=4 state=mature
        auth-key="8ac9dc7ecebfed9cd1030ae3b07b32e8e5cb98af"
        enc-key="8a8073a7afd0f74518c10438a0023e64cc660ed69845ca3c"
        add-lifetime=24m/30m use-lifetime=0s/0s lifebytes=0/0
        current-addtime=jan/28/2003 20:55:12
        current-usetime=jan/28/2003 20:55:12 current-bytes=512
[admin@WiFi] ip ipsec>
```

Flushing Installed SA Table

Command name: ***/ip ipsec installed-sa flush***

Description

Sometimes after incorrect/incomplete negotiations took place, it is required to flush manually the installed SA table so that SA could be renegotiated. This option is provided by the **flush** command.

Property Description

sa-type (*multiple choice: ah | all | esp*; default: **all**) - specifies SA types to flush

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **ah** - delete AH protocol SAs only
- **esp** - delete ESP protocol SAs only
- **all** - delete both ESP and AH protocols SAs

Example

To flush all the SAs installed:

```
[admin@MikroTik] ip ipsec installed-sa> flush
[admin@MikroTik] ip ipsec installed-sa> print
[admin@MikroTik] ip ipsec installed-sa>
```

Counters

Home menu level: ***/ip ipsec counters***

Property Description

in-accept (*read-only: integer*) - shows how many incoming packets were matched by accept policy

in-accept-isakmp (*read-only: integer*) - shows how many incoming UDP packets on port 500 were let through without matching a policy

in-decrypted (*read-only: integer*) - shows how many incoming packets were successfully decrypted

in-drop (*read-only: integer*) - shows how many incoming packets were matched by drop policy (or encrypt policy with level=require that does not have all necessary SAs)

in-drop-encrypted-expected (*read-only: integer*) - shows how many incoming packets were matched by encrypt policy and dropped because they were not encrypted

out-accept (*read-only: integer*) - shows how many outgoing packets were matched by accept policy (including the default "accept all" case)

out-accept-isakmp (*read-only: integer*) - shows how many locally originated UDP packets on source port 500 (which is how ISAKMP packets look) were let through without policy matching

out-drop (*read-only: integer*) - shows how many outgoing packets were matched by drop policy (or encrypt policy with level=require that does not have all necessary SAs)

out-encrypt (*read-only: integer*) - shows how many outgoing packets were encrypted successfully

Example

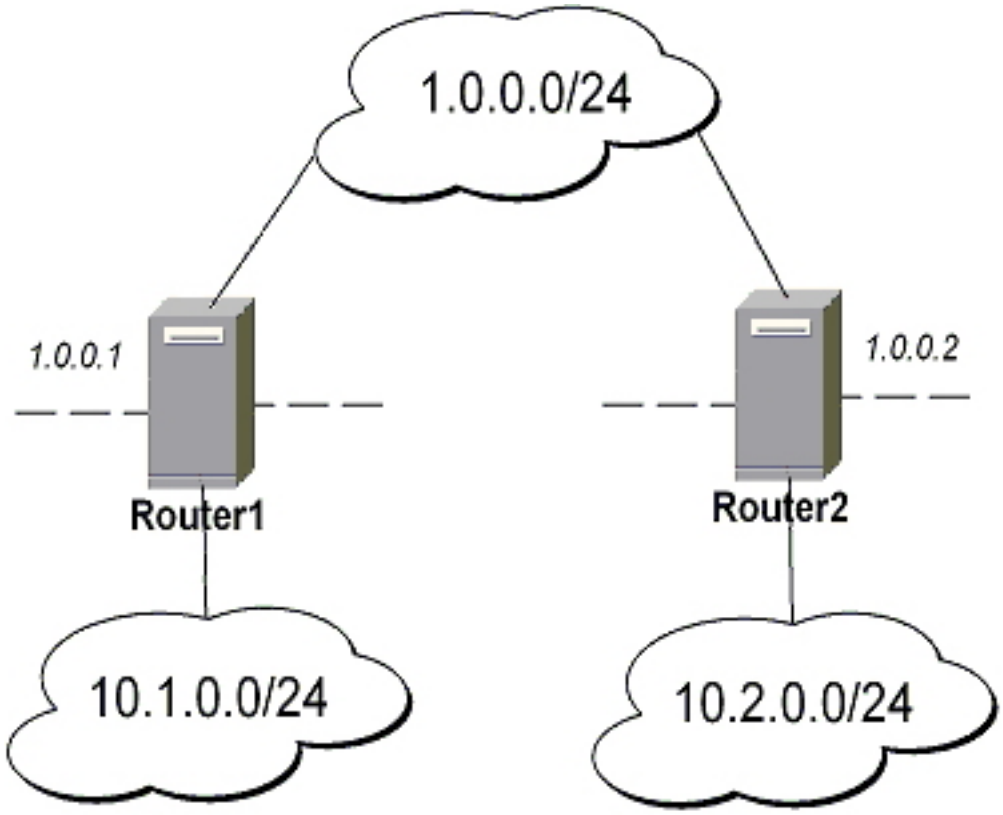
To view current statistics:

```
[admin@WiFi] ip ipsec> counters print
                    out-accept: 6
             out-accept-isakmp: 0
                      out-drop: 0
                   out-encrypt: 7
                     in-accept: 12
              in-accept-isakmp: 0
                       in-drop: 0
                   in-decrypted: 7
    in-drop-encrypted-expected: 0
[admin@WiFi] ip ipsec>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

General Information

MikroTik Router to MikroTik Router



- transport mode example using ESP with automatic keying
  - for **Router1**

```
[admin@Router1] > ip ipsec policy add sa-src-address=1.0.0.1 sa-dst-address=1.0.0.2 \
\... action=encrypt
[admin@Router1] > ip ipsec peer add address=1.0.0.2 \
\... secret="gvejimezyfopmekun"
```

  - for **Router2**

```
[admin@Router2] > ip ipsec policy add sa-src-address=1.0.0.2 sa-dst-address=1.0.0.1 \
\... action=encrypt
[admin@Router2] > ip ipsec peer add address=1.0.0.1 \
\... secret="gvejimezyfopmekun"
```

- transport mode example using ESP with automatic keying and automatic policy generating on Router 1 and static policy on Router 2
  - for **Router1**

```
[admin@Router1] > ip ipsec peer add address=1.0.0.0/24 \
\... secret="gvejimezyfopmekun" generate-policy=yes
```

  - for **Router2**

```
[admin@Router2] > ip ipsec policy add sa-src-address=1.0.0.2 sa-dst-address=1.0.0.1 \
\... action=encrypt
[admin@Router2] > ip ipsec peer add address=1.0.0.1 \
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
\... secret="gvejimezyfopmekun"
```

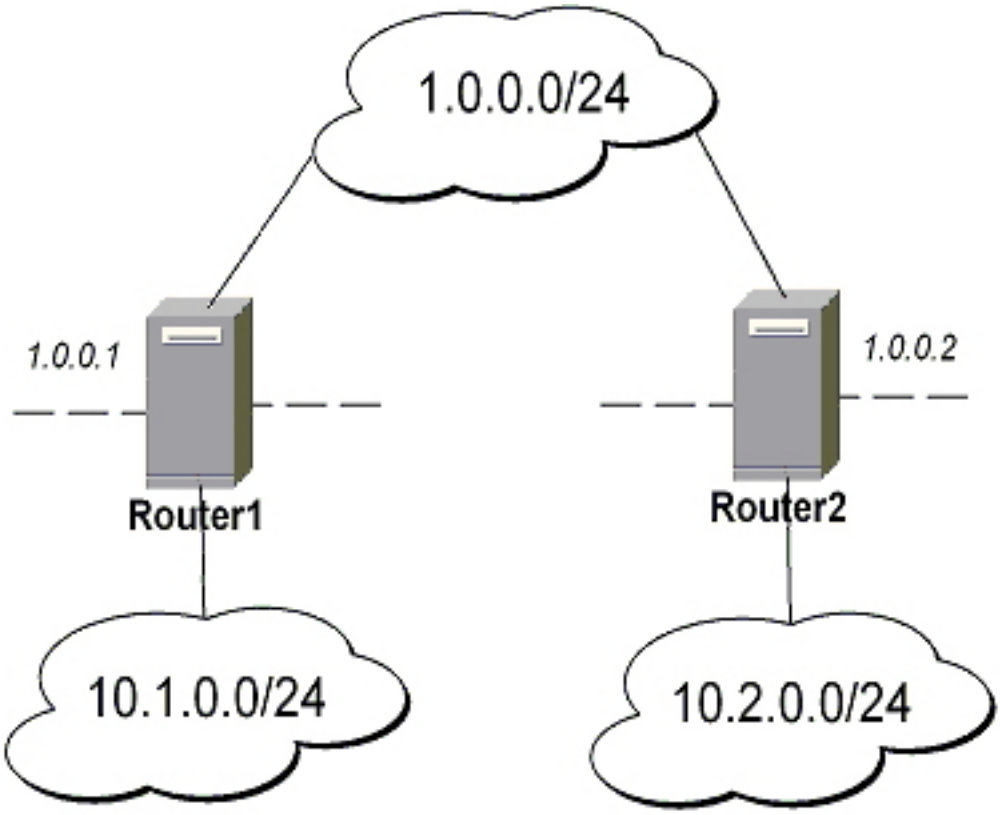
- tunnel mode example using AH with manual keying
  - for **Router1**

```
[admin@Router1] > ip ipsec manual-sa add name=ah-sa1 \
\... ah-spi=0x101/0x100 ah-key=abcfed
[admin@Router1] > ip ipsec policy add src-address=10.1.0.0/24 \
\... dst-address=10.2.0.0/24 action=encrypt ipsec-protocols=ah \
\... tunnel=yes sa-src=1.0.0.1 sa-dst=1.0.0.2 manual-sa=ah-sa1
```

  - for **Router2**

```
[admin@Router2] > ip ipsec manual-sa add name=ah-sa1 \
\... ah-spi=0x100/0x101 ah-key=abcfed
[admin@Router2] > ip ipsec policy add src-address=10.2.0.0/24 \
\... dst-address=10.1.0.0/24 action=encrypt ipsec-protocols=ah \
\... tunnel=yes sa-src=1.0.0.2 sa-dst=1.0.0.1 manual-sa=ah-sa1
```

IPsec Between two Masquerading MikroTik Routers



1. Add accept and masquerading rules in SRC-NAT
   - for **Router1**

```
[admin@Router1] > ip firewall nat \
\... add src-address=10.1.0.0/24 dst-address=10.2.0.0/24
[admin@Router1] > ip firewall nat add out-interface=public \
\... action=masquerade
```

   - for **Router2**

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@Router2] > ip firewall nat \
\... add src-address=10.2.0.0/24 dst-address=10.1.0.0/24
[admin@Router2] > ip firewall nat add out-interface=public \
\... action=masquerade
```

2. configure IPsec

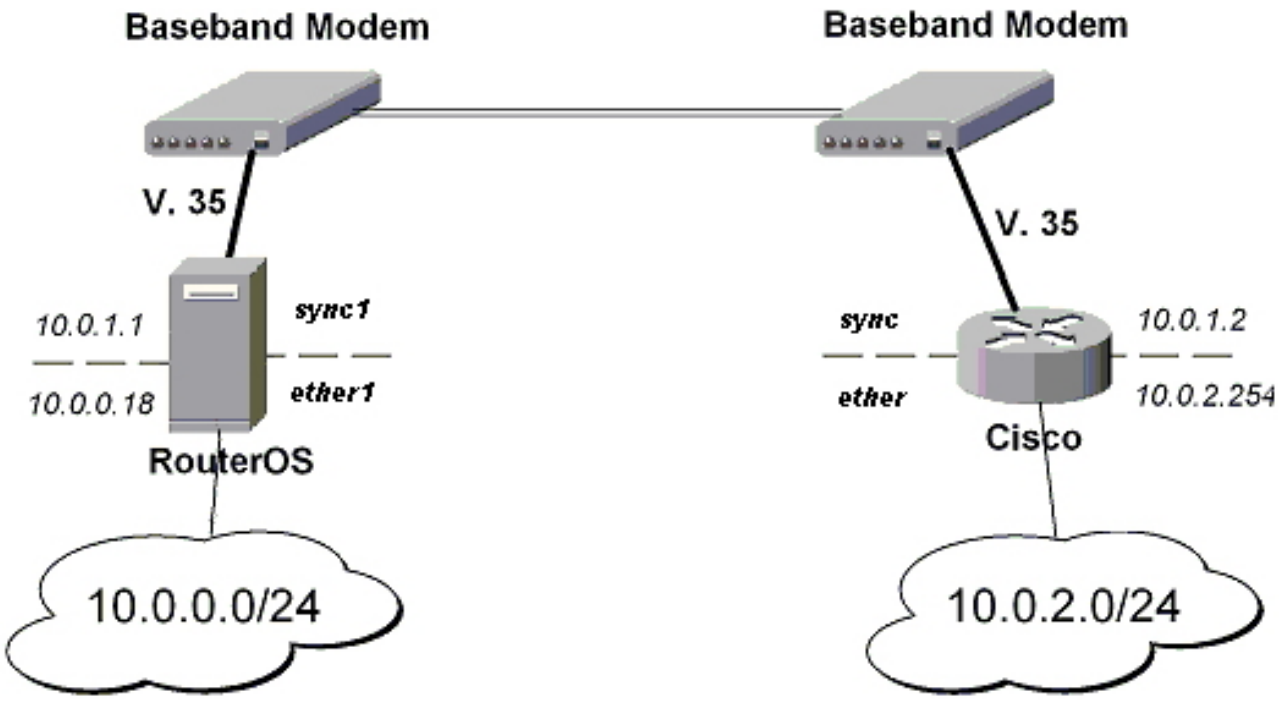
   - for **Router1**

```
[admin@Router1] > ip ipsec policy add src-address=10.1.0.0/24 \
\... dst-address=10.2.0.0/24 action=encrypt tunnel=yes \
\... sa-src-address=1.0.0.1 sa-dst-address=1.0.0.2
[admin@Router1] > ip ipsec peer add address=1.0.0.2 \
\... exchange-mode=aggressive secret="gvejimezyfopmekun"
```

   - for **Router2**

```
[admin@Router2] > ip ipsec policy add src-address=10.2.0.0/24 \
\... dst-address=10.1.0.0/24 action=encrypt tunnel=yes \
\... sa-src-address=1.0.0.2 sa-dst-address=1.0.0.1
[admin@Router2] > ip ipsec peer add address=1.0.0.1 \
\... exchange-mode=aggressive secret="gvejimezyfopmekun"
```

MikroTik router to CISCO Router



We will configure IPsec in tunnel mode in order to protect traffic between attached subnets.

1. Add peer (with phase1 configuration parameters), DES and SHA1 will be used to protect IKE traffic

   - for **MikroTik** router

```
[admin@MikroTik] > ip ipsec peer add address=10.0.1.2 \
\... secret="gvejimezyfopmekun" enc-algorithm=des
```

   - for **CISCO** router

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
! Configure ISAKMP policy (phase1 config, must match configuration
! of "/ip ipsec peer" on RouterOS). Note that DES is default
! encryption algorithm on Cisco. SHA1 is default authentication
! algorithm
crypto isakmp policy 9
  encryption des
  authentication pre-share
  group 2
  hash md5
  exit

! Add preshared key to be used when talking to RouterOS
crypto isakmp key gvejimezyfopmekun address 10.0.1.1 255.255.255.255
```

2. Set encryption proposal (phase2 proposal - settings that will be used to encrypt actual data) to use DES to encrypt data
    - for **MikroTik** router

```
[admin@MikroTik] > ip ipsec proposal set default enc-algorithms=des
```

    - for **CISCO** router

```
! Create IPsec transform set - transformations that should be applied to
! traffic - ESP encryption with DES and ESP authentication with SHA1
! This must match "/ip ipsec proposal"
crypto ipsec transform-set myset esp-des esp-sha-hmac
  mode tunnel
  exit
```

3. Add policy rule that matches traffic between subnets and requires encryption with ESP in tunnel mode
    - for **MikroTik** router

```
[admin@MikroTik] > ip ipsec policy add \
\... src-address=10.0.0.0/24 dst-address=10.0.2.0/24 action=encrypt \
\... tunnel=yes sa-src=10.0.1.1 sa-dst=10.0.1.2
```

    - for **CISCO** router

```
! Create access list that matches traffic that should be encrypted
access-list 101 permit ip 10.0.2.0 0.0.0.255 10.0.0.0 0.0.0.255
! Create crypto map that will use transform set "myset", use peer 10.0.1.1
! to establish SAs and encapsulate traffic and use access-list 101 to
! match traffic that should be encrypted
crypto map mymap 10 ipsec-isakmp
  set peer 10.0.1.1
  set transform-set myset
  set pfs group2
  match address 101
  exit
! And finally apply crypto map to serial interface:
interface Serial 0
  crypto map mymap
  exit
```

4. Testing the IPsec tunnel
    - on **MikroTik** router we can see installed SAs

```
[admin@MikroTik] ip ipsec installed-sa> print
Flags: A - AH, E - ESP, P - pfs, M - manual
  0 E   spi=9437482 direction=out src-address=10.0.1.1
        dst-address=10.0.1.2 auth-algorithm=sha1 enc-algorithm=des
        replay=4 state=mature
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
        auth-key="9cf2123b8b5add950e3e67b9eac79421d406aa09"
        enc-key="ffe7ec65b7a385c3" add-lifetime=24m/30m use-lifetime=0s/0s
        lifebytes=0/0 current-addtime=jul/12/2002 16:13:21
        current-usetime=jul/12/2002 16:13:21 current-bytes=71896
  1 E   spi=319317260 direction=in src-address=10.0.1.2
        dst-address=10.0.1.1 auth-algorithm=sha1 enc-algorithm=des
        replay=4 state=mature
        auth-key="7575f5624914dd312839694db2622a318030bc3b"
        enc-key="633593f809c9d6af" add-lifetime=24m/30m use-lifetime=0s/0s
        lifebytes=0/0 current-addtime=jul/12/2002 16:13:21
        current-usetime=jul/12/2002 16:13:21 current-bytes=0
[admin@MikroTik] ip ipsec installed-sa>
```

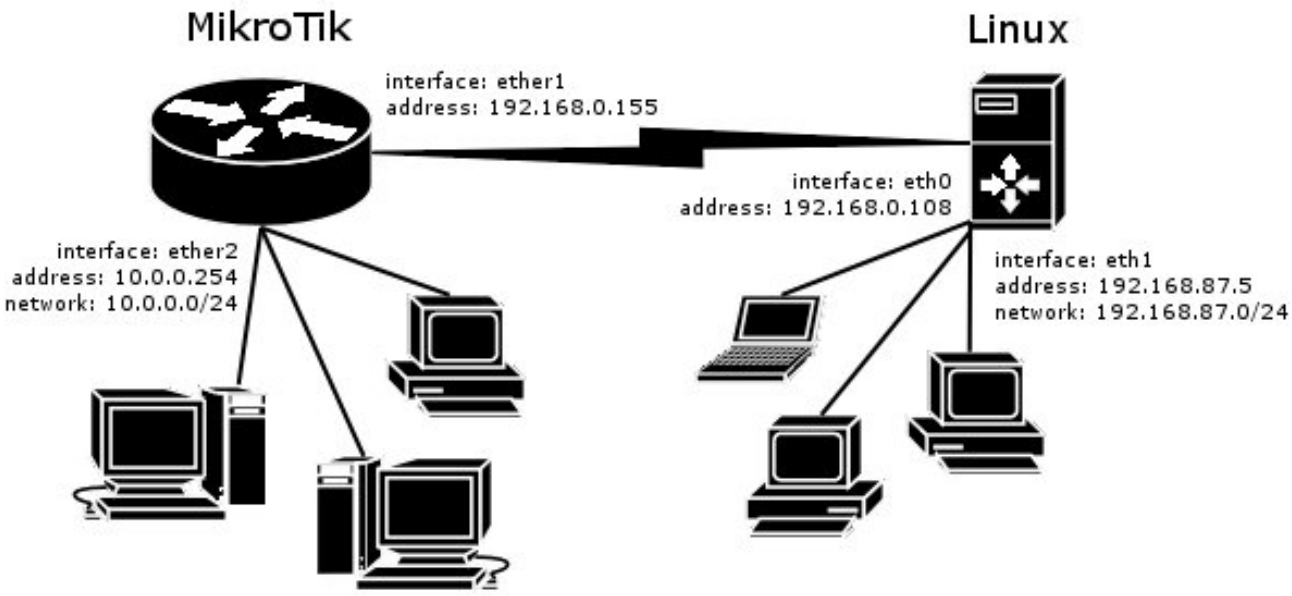
- on **CISCO** router

```
cisco# show interface Serial 0
interface: Serial1
    Crypto map tag: mymap, local addr. 10.0.1.2
   local  ident (addr/mask/prot/port): (10.0.2.0/255.255.255.0/0/0)
   remote ident (addr/mask/prot/port): (10.0.0.0/255.255.255.0/0/0)
   current_peer: 10.0.1.1
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 1810, #pkts encrypt: 1810, #pkts digest 1810
    #pkts decaps: 1861, #pkts decrypt: 1861, #pkts verify 1861
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0
     local crypto endpt.: 10.0.1.2, remote crypto endpt.: 10.0.1.1
     path mtu 1500, media mtu 1500
     current outbound spi: 1308650C
     inbound esp sas:
      spi: 0x90012A(9437482)
        transform: esp-des esp-sha-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2000, flow_id: 1, crypto map: mymap
        sa timing: remaining key lifetime (k/sec): (4607891/1034)
        IV size: 8 bytes
        replay detection support: Y
     inbound ah sas:
     inbound pcp sas:
     outbound esp sas:
      spi: 0x1308650C(319317260)
        transform: esp-des esp-sha-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2001, flow_id: 2, crypto map: mymap
        sa timing: remaining key lifetime (k/sec): (4607893/1034)
        IV size: 8 bytes
        replay detection support: Y
     outbound ah sas:
     outbound pcp sas:
```

MikroTik Router and Linux FreeS/WAN

In the test scenario we have 2 private networks: 10.0.0.0/24 connected to the MT and 192.168.87.0/24 connected to Linux. MT and Linux are connected together over the "public" network 192.168.0.0/24:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



- FreeS/WAN configuration:

```
config setup
    interfaces="ipsec0=eth0"
    klipsdebug=none
    plutodebug=all
    plutoload=%search
    plutostart=%search
    uniqueids=yes

conn %default
    keyingtries=0
    disablearrivalcheck=no
    authby=rsasig

conn mt
    left=192.168.0.108
    leftsubnet=192.168.87.0/24
    right=192.168.0.155
    rightsubnet=10.0.0.0/24
    authby=secret
    pfs=no
    auto=add
```

- **ipsec.secrets** config file:

```
192.168.0.108 192.168.0.155 : PSK "gvejimezyfopmekun"
```

- MikroTik Router configuration:

```
[admin@MikroTik] > /ip ipsec peer add address=192.168.0.108 \
\... secret="gvejimezyfopmekun" hash-algorithm=md5  enc-algorithm=3des \
\... dh-group=modp1024 lifetime=28800s

[admin@MikroTik] > /ip ipsec proposal auth-algorithms=md5 \
\... enc-algorithms=3des pfs-group=none

[admin@MikroTik] > /ip ipsec policy add sa-src-address=192.168.0.155 \
\... sa-dst-address=192.168.0.108 src-address=10.0.0.0/24 \
\... dst-address=192.168.87.0/24 tunnel=yes
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

IPIP Tunnel Interfaces

Document revision 1.1 (Fri Mar 05 08:25:43 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Quick Setup Guide
  Specifications
  Related Documents
  Additional Documents
IPIP Setup
  Description
  Property Description
  Notes
  Description

General Information

Summary

The IPIP tunneling implementation on the MikroTik RouterOS is RFC 2003 compliant. IPIP tunnel is a simple protocol that encapsulates IP packets in IP to make a tunnel between two routers. The IPIP tunnel interface appears as an interface under the interface list. Many routers, including Cisco and Linux based, support this protocol. This protocol makes multiple network schemes possible.

IP tunneling protocol adds the following possibilities to a network setups:

- to tunnel Intranets over the Internet
- to use it instead of source routing

Quick Setup Guide

To make an IPIP tunnel between 2 MikroTik routers with IP addresses **10.5.8.104** and **10.1.0.172**, using IPIP tunnel addresses 10.0.0.1 and 10.0.0.2, follow the next steps.

- Configuration on router with IP address **10.5.8.104**:
  1. Add an IPIP interface (by default, its name will be **ipip1**):

```
[admin@10.5.8.104] interface ipip> add local-address=10.5.8.104 \
remote-address=10.1.0.172 disabled=no
```

  2. Add an IP address to created **ipip1** interface:

```
[admin@10.5.8.104] ip address> add address=10.0.0.1/24 interface=ipip1
```

- Configuration on router with IP address **10.1.0.172**:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

1. Add an IPIP interface (by default, its name will be **ipip1**):

```
[admin@10.1.0.172] interface ipip> add local-address=10.1.0.172 \
remote-address=10.5.8.104 disabled=no
```

2. Add an IP address to created **ipip1** interface:

```
[admin@10.1.0.172] ip address> add address=10.0.0.2/24 interface=ipip1
```

Specifications

Packages required: ***system***
License required: ***level1 (limited to 1 tunnel), level3 (200 tunnels), level5 (unlimited)***
Home menu level: ***/interface ipip***
Standards and Technologies: ***IPIP (RFC 2003)***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Device Driver List***
- ***IP Addresses and ARP***
- ***Log Management***

Additional Documents

- ***http://www.ietf.org/rfc/rfc1853.txt?number=1853***
- ***http://www.ietf.org/rfc/rfc2003.txt?number=2003***
- ***http://www.ietf.org/rfc/rfc1241.txt?number=1241***

IPIP Setup

Home menu level: ***/interface ipip***

Description

An IPIP interface should be configured on two routers that have the possibility for an IP level connection and are ***RFC 2003*** compliant. The IPIP tunnel may run over any connection that transports IP. Each IPIP tunnel interface can connect with one remote router that has a corresponding interface configured. An unlimited number of IPIP tunnels may be added to the router. For more details on IPIP tunnels, see ***RFC 2003***.

Property Description

name (*name*; default: **ipipN**) - interface name for reference

mtu (*integer*; default: **1480**) - Maximum Transmission Unit. Should be set to 1480 bytes to avoid fragmentation of packets. May be set to 1500 bytes if mtu path discovery is not working properly

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

on links

local-address (*IP address*) - local address on router which sends IPIP traffic to the remote host

remote-address (*IP address*) - the IP address of the remote host of the IPIP tunnel - may be any RFC 2003 compliant router

Notes

Use **/ip address add** command to assign an **IP address** to the IPIP interface.

There is no authentication or 'state' for this interface. The bandwidth usage of the interface may be monitored with the **monitor** feature from the **interface** menu.

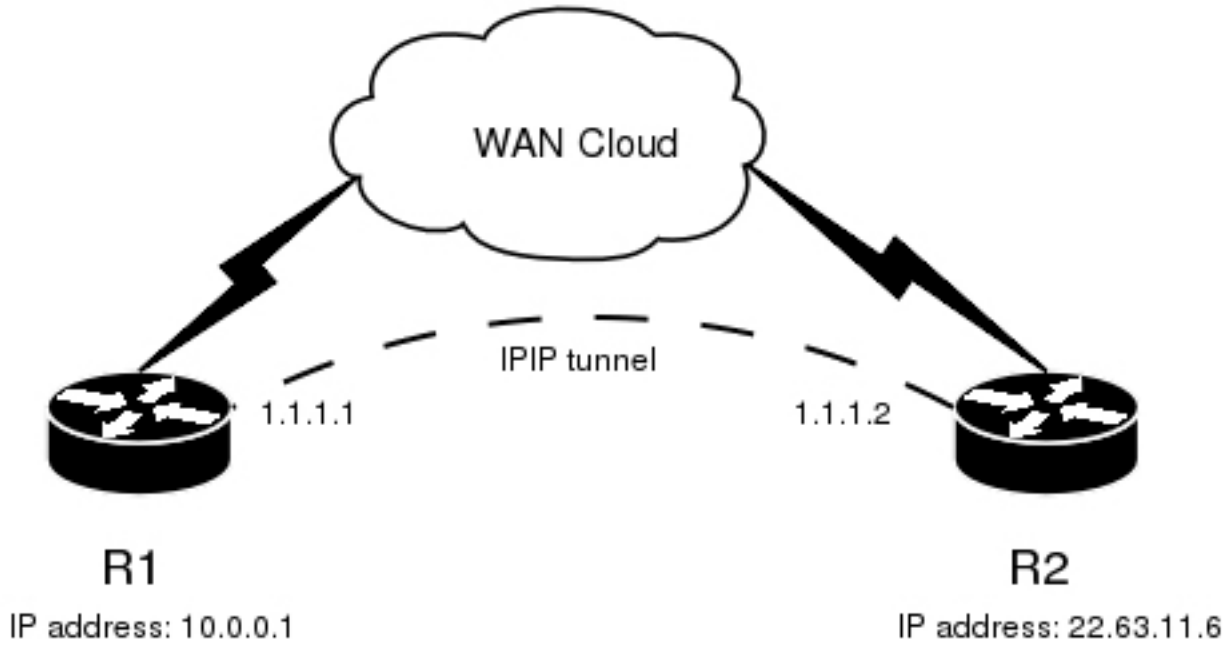
MikroTik RouterOS IPIP implementation has been tested with Cisco 1005. The sample of the Cisco 1005 configuration is given below:

```
interface Tunnel0
 ip address 10.3.0.1 255.255.255.0
 tunnel source 10.0.0.171
 tunnel destination 10.0.0.204
 tunnel mode ipip
```

General Information

Description

Suppose we want to add an IPIP tunnel between routers **R1** and **R2**:



At first, we need to configure IPIP interfaces and then add **IP addresses** to them.

The configuration for router **R1** is as follows:

```
[admin@MikroTik] interface ipip> add
local-address: 10.0.0.1
remote-address: 22.63.11.6
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] interface ipip> print
Flags: X - disabled, R - running
  #    NAME                                MTU   LOCAL-ADDRESS   REMOTE-ADDRESS
  0 X  ipip1                               1480  10.0.0.1        22.63.11.6

[admin@MikroTik] interface ipip> en 0
[admin@MikroTik] interface ipip> /ip address add address 1.1.1.1/24 interface=ipip1
```

The configuration of the **R2** is shown below:

```
[admin@MikroTik] interface ipip> add local-address=22.63.11.6 remote-address=10.
0.0.1
[admin@MikroTik] interface ipip> print
Flags: X - disabled, R - running
  #    NAME                                MTU   LOCAL-ADDRESS   REMOTE-ADDRESS
  0 X  ipip1                               1480  22.63.11.6      10.0.0.1

[admin@MikroTik] interface ipip> enable 0
[admin@MikroTik] interface ipip> /ip address add address 1.1.1.2/24 interface=ipip1
```

Now both routers can ping each other:

```
[admin@MikroTik] interface ipip> /ping 1.1.1.2
1.1.1.2 64 byte ping: ttl=64 time=24 ms
1.1.1.2 64 byte ping: ttl=64 time=19 ms
1.1.1.2 64 byte ping: ttl=64 time=20 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 19/21.0/24 ms
[admin@MikroTik] interface ipip>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

L2TP Interface

Document revision 1.1 (Fri Mar 05 08:26:01 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Quick Setup Guide
  Specifications
  Related Documents
  Description
L2TP Client Setup
  Property Description
  Example
Monitoring L2TP Client
  Property Description
  Example
L2TP Server Setup
  Description
  Property Description
  Example
L2TP Server Users
  Description
  Property Description
  Example
L2TP Application Examples
  Router-to-Router Secure Tunnel Example
  Connecting a Remote Client via L2TP Tunnel
  L2TP Setup for Windows
Troubleshooting
  Description

General Information

Summary

L2TP (Layer 2 Tunnel Protocol) supports encrypted tunnels over IP. The MikroTik RouterOS implementation includes support for both L2TP client and server.

General applications of L2TP tunnels include:

- secure router-to-router tunnels over the Internet
- linking (bridging) local Intranets or LANs (in cooperation with EoIP)
- extending PPP user connections to a remote location (for example, to separate authentication and Internet access points for ISP)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- accessing an Intranet/LAN of a company for remote (mobile) clients (employees)

Each L2TP connection is composed of a server and a client. The MikroTik RouterOS may function as a server or client or, for various configurations, it may be the server for some connections and client for other connections.

Quick Setup Guide

To make a L2TP tunnel between 2 MikroTik routers with IP addresses **10.5.8.104** (L2TP server) and **10.1.0.172** (L2TP client), follow the next steps.

- Configuration on L2TP server router:
  1. Add a L2TP user:

```
[admin@L2TP-Server] ppp secret> add name=james password=pass \
\... local-address=10.0.0.1 remote-address=10.0.0.2
```

  2. Enable the L2TP server

```
[admin@L2TP-Server] interface l2tp-server server> set enabled=yes
```

- Configuration on L2TP client router:
  1. Add a L2TP client:

```
[admin@L2TP-Client] interface l2tp-client> add user=james password=pass \
\... connect-to=10.5.8.104
```

Specifications

Packages required: ***ppp***
License required: ***level1 (limited to 1 tunnel), level3 (limited to 200 tunnels), level5***
Home menu level: ***/interface l2tp-server, /interface l2tp-client***
Standards and Technologies: ***L2TP (RFC 2661)***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***IP Addresses and ARP***
- ***AAA***
- ***EoIP Tunnel Interface***
- ***IP Security***

Description

L2TP is a secure tunnel protocol for transporting IP traffic using PPP. L2TP encapsulates PPP in virtual lines that run over IP, Frame Relay and other protocols (that are not currently supported by MikroTik RouterOS). L2TP incorporates PPP and MPPE (Microsoft Point to Point Encryption) to make encrypted links. The purpose of this protocol is to allow the Layer 2 and PPP endpoints to

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

reside on different devices interconnected by a packet-switched network. With L2TP, a user has a Layer 2 connection to an access concentrator - **LAC** (e.g., modem bank, ADSL DSLAM, etc.), and the concentrator then tunnels individual PPP frames to the Network Access Server - **NAS**. This allows the actual processing of PPP packets to be divorced from the termination of the Layer 2 circuit. From the user's perspective, there is no functional difference between having the L2 circuit terminate in a NAS directly or using L2TP.

It may also be useful to use L2TP just as any other tunneling protocol with or without encryption. The L2TP standard says that the most secure way to encrypt data is using L2TP over IPsec (**Note** that it is default mode for Microsoft L2TP client) as all L2TP control and data packets for a particular tunnel appear as homogeneous UDP/IP data packets to the IPsec system.

L2TP includes PPP authentication and accounting for each L2TP connection. Full authentication and accounting of each connection may be done through a RADIUS client or locally.

MPPE 40bit RC4 and MPPE 128bit RC4 encryption are supported.

L2TP traffic uses UDP protocol for both control and data packets. UDP port 1701 is used only for link establishment, further traffic is using any available UDP port (which may or may not be 1701). This means that L2TP can be used with most firewalls and routers (even with NAT) by enabling UDP traffic to be routed through the firewall or router.

L2TP Client Setup

Home menu level: ***/interface l2tp-client***

Property Description

name (*name*; default: **l2tp-outN**) - interface name for reference

mtu (*integer*; default: **1460**) - Maximum Transmission Unit. The optimal value is the MTU of the interface the tunnel is working over decreased by 40 (so, for 1500-byte Ethernet link, set the MTU to 1460 to avoid fragmentation of packets)

mru (*integer*; default: **1460**) - Maximum Receive Unit. The optimal value is the MRU of the interface the tunnel is working over decreased by 40 (so, for 1500-byte Ethernet link, set the MRU to 1460 to avoid fragmentation of packets)

connect-to (*IP address*) - The IP address of the L2TP server to connect to

user (*text*) - user name to use when logging on to the remote server

password (*text*; default: **""**) - user password to use when logging to the remote server

profile (*name*; default: **default**) - profile to use when connecting to the remote server

allow (*multiple choice: mschap2, mschap1, chap, pap*; default: **mschap2, mschap1, chap, pap**) - the protocol to allow the client to use for authentication

add-default-route (*yes* | *no*; default: **no**) - whether to use the server which this client is connected to as its default router (gateway)

Example

To set up L2TP client named **test2** using username **john** with password **john** to connect to the **10.1.1.12** L2TP server and use it as the default gateway:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] interface l2tp-client> add name=test2 connect-to=10.1.1.12 \
\... user=john add-default-route=yes password=john
[admin@MikroTik] interface l2tp-client> print
Flags: X - disabled, R - running
  0 X  name="test2" mtu=1460 mru=1460 connect-to=10.1.1.12 user="john"
       password="john" profile=default add-default-route=yes


[admin@MikroTik] interface l2tp-client> enable 0
```

Monitoring L2TP Client

Command name: ***/interface l2tp-client monitor***

Property Description

status (*text*) - status of the client

- **Dialing** - attempting to make a connection
- **Verifying password...** - connection has been established to the server, password verification in progress
- **Connected** - self-explanatory
- **Terminated** - interface is not enabled or the other side will not establish a connection uptime (time) - connection time displayed in days, hours, minutes and seconds

encoding (*text*) - encryption and encoding (if asymmetric, separated with '/') being used in this connection

Example

Example of an established connection

```
[admin@MikroTik] interface l2tp-client> monitor test2
      status: "connected"
      uptime: 4m27s
    encoding: "MPPE128 stateless"
[admin@MikroTik] interface l2tp-client>
```

L2TP Server Setup

Home menu level: ***/interface l2tp-server server***

Description

The L2TP server creates a dynamic interface for each connected L2TP client. The L2TP connection count from clients depends on the license level you have. Level1 license allows 1 L2TP client, Level3 or Level4 licenses up to 200 clients, and Level5 or Level6 licenses do not have L2TP client limitations.

To create L2TP users, you should consult the ***PPP secret*** and ***PPP Profile*** manuals. It is also possible to use the MikroTik router as a RADIUS client to register the L2TP users, see the ***manual*** how to do it.

Property Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

enabled (*yes* | *no*; default: **no**) - defines whether L2TP server is enabled or not

mtu (*integer*; default: **1460**) - Maximum Transmission Unit. The optimal value is the MTU of the interface the tunnel is working over decreased by 40 (so, for 1500-byte Ethernet link, set the MTU to 1460 to avoid fragmentation of packets)

mru (*integer*; default: **1460**) - Maximum Receive Unit. The optimal value is the MRU of the interface the tunnel is working over decreased by 40 (so, for 1500-byte Ethernet link, set the MRU to 1460 to avoid fragmentation of packets)

authentication (*multiple choice: pap* | *chap* | *mschap1* | *mschap2*; default: **mschap2**) - authentication algorithm

default-profile - default profile to use

Example

To enable L2TP server:

```
[admin@MikroTik] interface l2tp-server server> set enabled=yes
[admin@MikroTik] interface l2tp-server server> print
            enabled: yes
                mtu: 1460
                mru: 1460
     authentication: mschap2
    default-profile: default
[admin@MikroTik] interface l2tp-server server>
```

L2TP Server Users

Home menu level: ***/interface l2tp-server***

Description

There are two types of items in L2TP server configuration - static users and dynamic connections. A dynamic connection can be established if the user database or the **default-profile** has its **local-address** and **remote-address** set correctly. When static users are added, the default profile may be left with its default values and only PPP user (in **/ppp secret**) should be configured. **Note** that in both cases PPP users must be configured properly.

Property Description

name (*name*) - interface name

user (*text*) - the name of the user that is configured statically or added dynamically

mtu - shows client's MTU

client-address - shows the IP of the connected client

uptime - shows how long the client is connected

encoding (*text*) - encryption and encoding (if asymmetric, separated with '/') being used in this connection

Example

To add a static entry for **ex1** user:

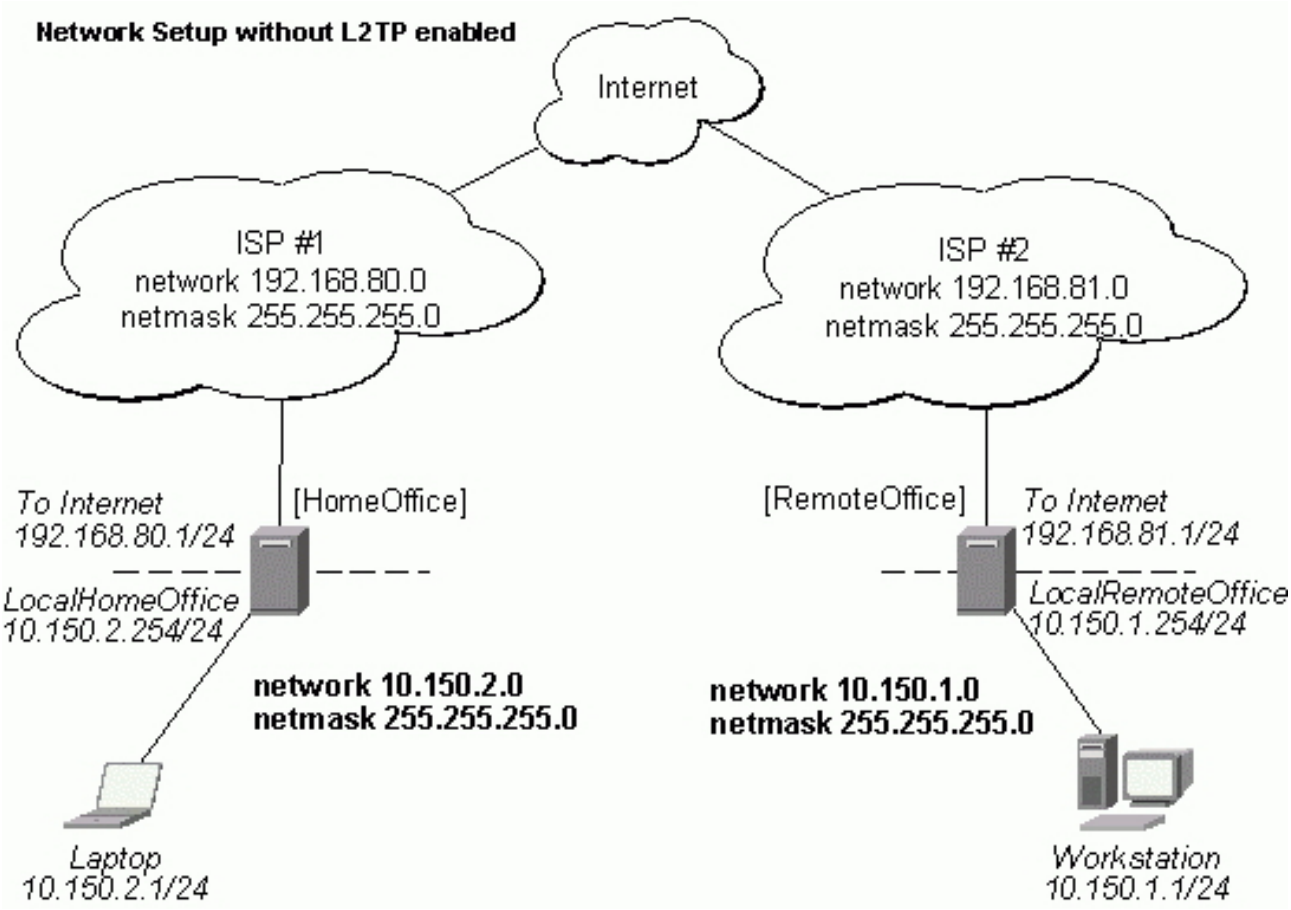
Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] interface l2tp-server> add user=ex1
[admin@MikroTik] interface l2tp-server> print
Flags: X - disabled, D - dynamic, R - running
  #     NAME                USER          MTU   CLIENT-ADDRESS  UPTIME   ENC...
  0  DR <l2tp-ex>           ex            1460  10.0.0.202      6m32s    none
  1     l2tp-in1            ex1
[admin@MikroTik] interface l2tp-server>
```

In this example an already connected user **ex** is shown besides the one we just added.

L2TP Application Examples

Router-to-Router Secure Tunnel Example



There are two routers in this example:

- [HomeOffice]
  Interface LocalHomeOffice 10.150.2.254/24
  Interface ToInternet 192.168.80.1/24
- [RemoteOffice]
  Interface ToInternet 192.168.81.1/24
  Interface LocalRemoteOffice 10.150.1.254/24

Each router is connected to a different ISP. One router can access another router through the Internet.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

On the L2TP server a user must be set up for the client:

```
[admin@HomeOffice] ppp secret> add name=ex service=l2tp password=lkjrht
local-address=10.0.103.1 remote-address=10.0.103.2
[admin@HomeOffice] ppp secret> print detail
Flags: X - disabled
  0   name="ex" service=l2tp caller-id="" password="lkjrht" profile=default
      local-address=10.0.103.1 remote-address=10.0.103.2 routes==""

[admin@HomeOffice] ppp secret>
```

Then the user should be added in the L2TP server list:

```
[admin@HomeOffice] interface l2tp-server> add user=ex
[admin@HomeOffice] interface l2tp-server> print
Flags: X - disabled, D - dynamic, R - running
  #     NAME                USER         MTU   CLIENT-ADDRESS  UPTIME   ENC...
  0     l2tp-in1            ex
[admin@HomeOffice] interface l2tp-server>
```

And finally, the server must be enabled:

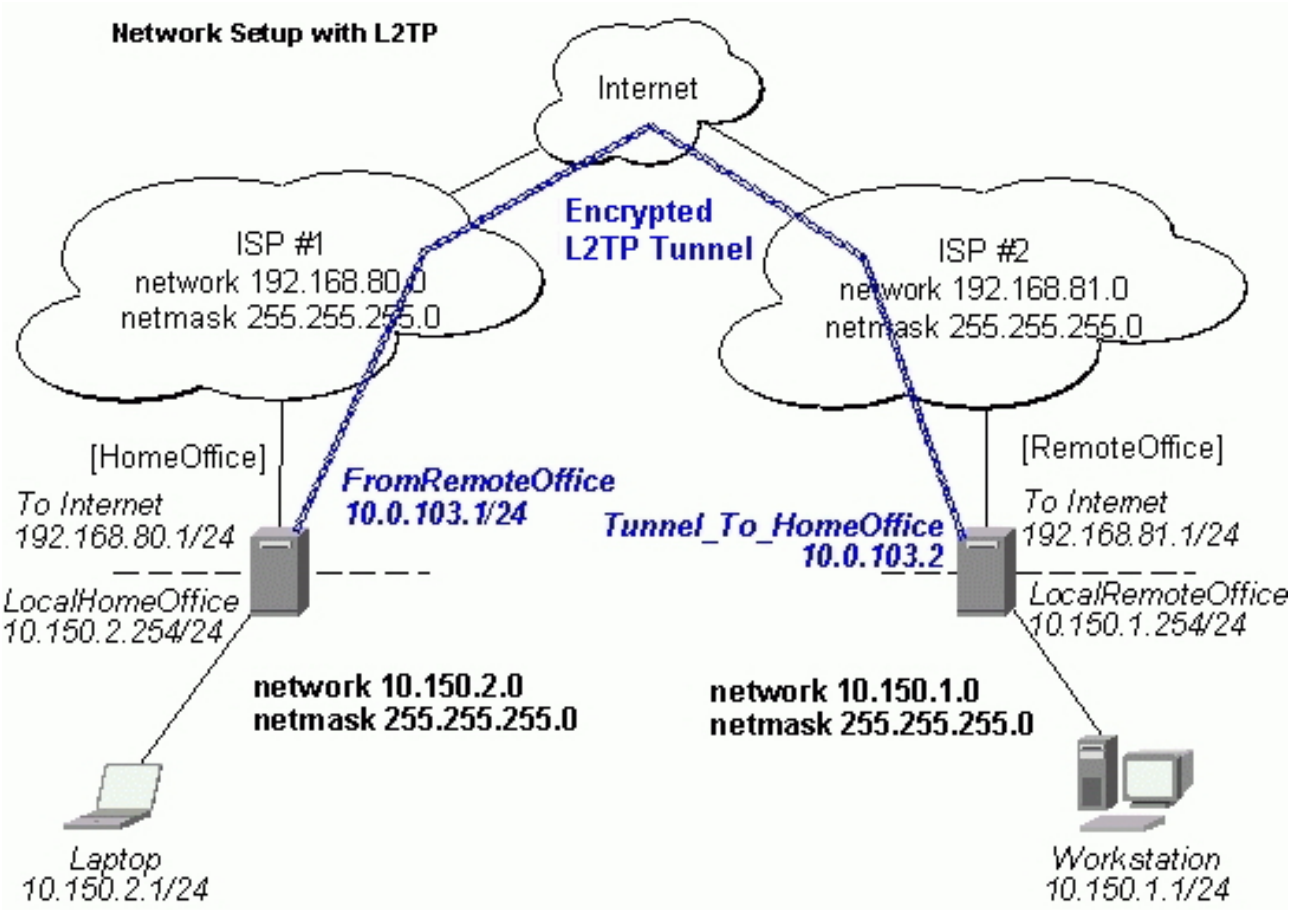
```
[admin@HomeOffice] interface l2tp-server server> set enabled=yes
[admin@HomeOffice] interface l2tp-server server> print
            enabled: yes
                mtu: 1460
                mru: 1460
     authentication: mschap2
    default-profile: default
[admin@HomeOffice] interface l2tp-server server>
```

Add a L2TP client to the RemoteOffice router:

```
[admin@RemoteOffice] interface l2tp-client> add connect-to=192.168.80.1 user=ex \
\... password=lkjrht disabled=no
[admin@RemoteOffice] interface l2tp-client> print
Flags: X - disabled, R - running
  0  R name="l2tp-out1" mtu=1460 mru=1460 connect-to=192.168.80.1 user="ex"
       password="lkjrht" profile=default add-default-route=no


[admin@RemoteOffice] interface l2tp-client>
```

Thus, a L2TP tunnel is created between the routers. This tunnel is like an Ethernet point-to-point connection between the routers with IP addresses 10.0.103.1 and 10.0.103.2 at each router. It enables 'direct' communication between the routers over third party networks.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



To route the local Intranets over the L2TP tunnel you need to add these routes:

```
[admin@HomeOffice] > ip route add dst-address 10.150.1.0/24 gateway 10.0.103.2
[admin@RemoteOffice] > ip route add dst-address 10.150.2.0/24 gateway 10.0.103.1
```

On the L2TP server it can alternatively be done using **routes** parameter of the user configuration:

```
[admin@HomeOffice] ppp secret> print detail
Flags: X - disabled
  0   name="ex" service=l2tp caller-id="" password="lkjrht" profile=default
      local-address=10.0.103.1 remote-address=10.0.103.2 routes==""

[admin@HomeOffice] ppp secret> set 0 routes="10.150.1.0/24 10.0.103.2 1"
[admin@HomeOffice] ppp secret> print detail
Flags: X - disabled
  0   name="ex" service=l2tp caller-id="" password="lkjrht" profile=default
      local-address=10.0.103.1 remote-address=10.0.103.2
      routes="10.150.1.0/24 10.0.103.2 1"

[admin@HomeOffice] ppp secret>
```

Test the L2TP tunnel connection:

```
[admin@RemoteOffice]> /ping 10.0.103.1
10.0.103.1 pong: ttl=255 time=3 ms
10.0.103.1 pong: ttl=255 time=3 ms
10.0.103.1 pong: ttl=255 time=3 ms
ping interrupted
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 3/3.0/3 ms
```

Test the connection through the L2TP tunnel to the LocalHomeOffice interface:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

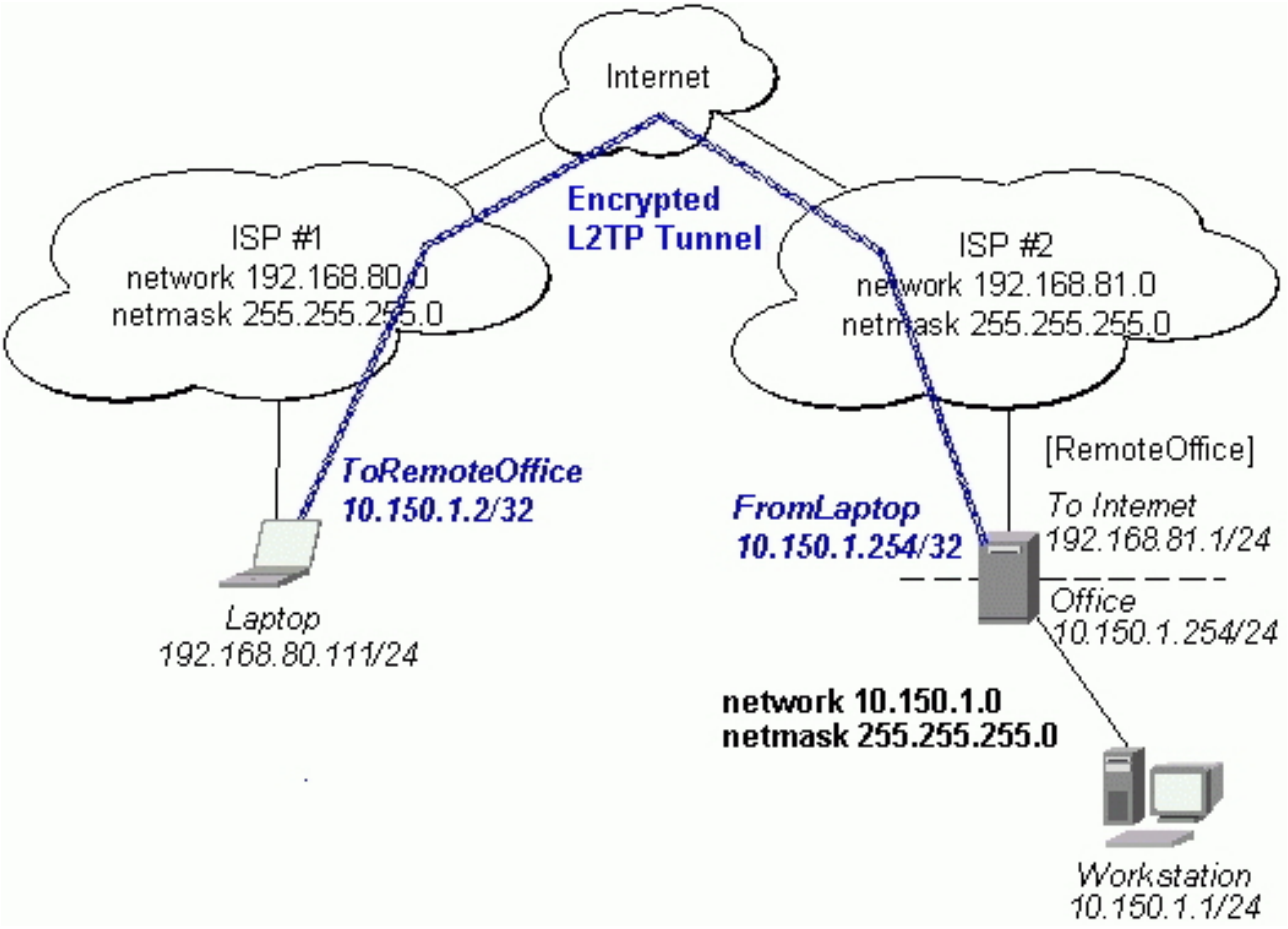
```
[admin@RemoteOffice]> /ping 10.150.2.254
10.150.2.254 pong: ttl=255 time=3 ms
10.150.2.254 pong: ttl=255 time=3 ms
10.150.2.254 pong: ttl=255 time=3 ms
ping interrupted
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 3/3.0/3 ms
```

To bridge a LAN over this secure tunnel, please see the example in the 'EoIP' section of the manual. To set the maximum speed for traffic over this tunnel, please consult the 'Queues' section.

Connecting a Remote Client via L2TP Tunnel

The following example shows how to connect a computer to a remote office network over L2TP encrypted tunnel giving that computer an IP address from the same network as the remote office has (without need of bridging over EoIP tunnels).

Please, consult the respective manual on how to set up a L2TP client with the software you are using.



The router in this example:

- [RemoteOffice]
  Interface ToInternet 192.168.81.1/24
  Interface Office 10.150.1.254/24

The client computer can access the router through the Internet.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

On the L2TP server a user must be set up for the client:

```
[admin@RemoteOffice] ppp secret> add name=ex service=l2tp password=lkjrht
local-address=10.150.1.254 remote-address=10.150.1.2
[admin@RemoteOffice] ppp secret> print detail
Flags: X - disabled
  0   name="ex" service=l2tp caller-id="" password="lkjrht" profile=default
      local-address=10.150.1.254 remote-address=10.150.1.2 routes==""

[admin@RemoteOffice] ppp secret>
```

Then the user should be added in the L2TP server list:

```
[admin@RemoteOffice] interface l2tp-server> add name=FromLaptop user=ex
[admin@RemoteOffice] interface l2tp-server> print
Flags: X - disabled, D - dynamic, R - running
  #     NAME                USER         MTU   CLIENT-ADDRESS  UPTIME   ENC...
  0     FromLaptop          ex
[admin@RemoteOffice] interface l2tp-server>
```

And the server must be enabled:

```
[admin@RemoteOffice] interface l2tp-server server> set enabled=yes
[admin@RemoteOffice] interface l2tp-server server> print
            enabled: yes
                mtu: 1460
                mru: 1460
     authentication: mschap2
    default-profile: default
[admin@RemoteOffice] interface l2tp-server server>
```

Finally, the proxy APR must be enabled on the 'Office' interface:

```
[admin@RemoteOffice] interface ethernet> set Office arp=proxy-arp
[admin@RemoteOffice] interface ethernet> print
Flags: X - disabled, R - running
  #    NAME                MTU   MAC-ADDRESS       ARP
  0  R ToInternet          1500  00:30:4F:0B:7B:C1 enabled
  1  R Office              1500  00:30:4F:06:62:12 proxy-arp
[admin@RemoteOffice] interface ethernet>
```

L2TP Setup for Windows

Microsoft provides L2TP client support for Windows XP, 2000, NT4, ME and 98. Windows 2000 and XP include support in the Windows setup or automatically install L2TP. For 98, NT and ME, installation requires a download from Microsoft (L2TP/IPsec VPN Client).

For more information, see:

Microsoft L2TP/IPsec VPN Client Microsoft L2TP/IPsec VPN Client

On Windows 2000, L2TP setup without IPsec requires editing registry:

Disabling IPsec for the Windows 2000 Client

Disabling IPSEC Policy Used with L2TP

Troubleshooting

Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **I use firewall and I cannot establish L2TP connection**
  Make sure UDP connections can pass through both directions between your sites.
- **My Windows L2TP/IPsec VPN Client fails to connect to L2TP server with "Error 789" or "Error 781"**
  The error messages 789 and 781 occur when IPsec is not configured properly on both ends. See the respective documentation on how to configure IPsec in the Microsoft L2TP/IPsec VPN Client and in the MikroTik RouterOS. If you do not want to use IPsec, it can be easily switched off on the client side. Note: if you are using Windows 2000, you need to edit system registry using regedt32.exe or regedit.exe. Add the following registry value to **HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Rasman\Parameters**:

```
Value Name: ProhibitIpSec
Data Type: REG_DWORD
Value: 1
```

You must restart the Windows 2000 for the changes to take effect

For more information on configuring Windows 2000, see:

- ***Configuring Cisco IOS and Windows 2000 Clients for L2TP Using Microsoft IAS***
- ***Disabling IPSEC Policy Used with L2TP***
- ***How to Configure a L2TP/IPsec Connection Using Pre-shared Key Authentication***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

PPPoE

Document revision 1.4 (Fri Apr 30 06:43:11 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Quick Setup Guide
  Specifications
  Related Documents
  Additional Documents
PPPoE Client Setup
  Description
  Property Description
  Notes
  Example
Monitoring PPPoE Client
  Property Description
  Example
PPPoE Server Setup (Access Concentrator)
  Description
  Property Description
  Notes
  Example
PPPoE Server Users
  Property Description
  Example
Application Examples
  PPPoE in a multipoint wireless 802.11g network
Troubleshooting
  Description

General Information

Summary

The PPPoE (Point to Point Protocol over Ethernet) protocol provides extensive user management, network management and accounting benefits to ISPs and network administrators. Currently PPPoE is used mainly by ISPs to control client connections for xDSL and cable modems as well as plain Ethernet networks. PPPoE is an extension of the standard Point to Point Protocol (PPP). The difference between them is expressed in transport method: PPPoE employs Ethernet instead of modem connection.

Generally speaking, PPPoE is used to hand out IP addresses to clients based on the user (and workstation, if desired) authentication as opposed to workstation only authentication, when static IP

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

addresses or DHCP are used. It is adviced not to use static IP addresses or DHCP on the same interfaces as PPPoE for security reasons.

MikroTik RouterOS can act as a RADIUS client - you can use a RADIUS server to authenticate PPPoE clients and use accounting for them.

A PPPoE connection is composed of a client and an access concentrator (server). The client may be a Windows computer that has the PPPoE client protocol installed. The MikroTik RouterOS supports both - client and access concentrator implementations of PPPoE. The PPPoE client and server work over any Ethernet level interface on the router - wireless 802.11 (Aironet, Cisco, WaveLan, Prism, Atheros), 10/100/1000 Mbit/s Ethernet, RadioLan and EoIP (Ethernet over IP tunnel). No encryption, MPPE 40bit RSA and MPPE 128bit RSA encryption is supported.

Note that when RADIUS server is authenticating a user with CHAP, MS-CHAPv1, MS-CHAPv2, it does not use shared secret, it is used only in authentication reply. So if you have a wrong shared secret, RADIUS server will accept the request. You can use **/radius monitor** command to see **bad-replies** parameter. This value should increase whenever a client tries to connect.

Supported connections

- MikroTik RouterOS PPPoE client to any PPPoE server (access concentrator)
- MikroTik RouterOS server (access concentrator) to multiple PPPoE clients (clients are avaliable for almost all operating systems and some routers)

Quick Setup Guide

- To configure MikroTik RouterOS to be a PPPoE client
  1. Just add a pppoe-client:

```
/interface pppoe-client add name=pppoe-user-mike user=mike password=123 interface=wlan1
\
\... service-name=internet disabled=no
```

- To configure MikroTik RouterOS to be an Access Concentrator (PPPoE Server)
  1. Add an address pool for the clients from **10.1.1.62** to **10.1.1.72**, called pppoe-pool:

```
/ip pool add name="pppoe-pool" ranges=10.1.1.62-10.1.1.72
```

  2. Add PPP profile, called **pppoe-profile** where **local-address** will be the router's address and clients will have an address from **pppoe-pool**:

```
/ppp profile add name="pppoe-profile" local-address=10.1.1.1 remote-address=pppoe-pool
```

  3. Add a user with username **mike** and password **123**:

```
/ppp secret add name=mike password=123 service=pppoe profile=pppoe-profile
```

  4. Now add a pppoe server:

```
/interface pppoe-server server add service-name=internet interface=wlan1 \
\... default-profile=pppoe-profile
```

Specifications

Packages required: ***ppp***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

License required: ***level1 (limited to 1 interface), level3 (limited to 200 interfaces), level4 (limited to 200 interfaces), level5 (limited to 500 interfaces), level6 (unlimited)***
Home menu level: ***/interface pppoe-server, /interface pppoe-client***
Standards and Technologies: ***PPPoE (RFC 2516)***
Hardware usage: *PPPoE server may require additional RAM (uses approx. 9KiB (plus extra 10KiB, if data rate limitation is used) for each connection) and CPU power. Supports maximum of 65535 connections*

Related Documents

- ***Package Management***
- ***IP Addresses and ARP***
- ***Log Management***

Additional Documents

Links for PPPoE documentation:

- ***http://www.faqs.org/rfcs/rfc2516.html***

PPPoE Clients:

- RASPPPoE for Windows 95, 98, 98SE, ME, NT4, 2000, XP, .NET
  http://www.raspppoe.com/

PPPoE Client Setup

Home menu level: ***/interface pppoe-client***

Description

The PPPoE client supports high-speed connections. It is fully compatible with the MikroTik PPPoE server (access concentrator).

Note for Windows. Some connection instructions may use the form where the "phone number" us "MikroTik_AC\mt1" to indicate that "MikroTik_AC" is the access concentrator name and "mt1" is the service name.

Property Description

name (*name*; default: **pppoe-out1**) - name of the PPPoE interface

interface (*name*) - interface the PPPoE server can be connected through

mtu (*integer*; default: **1480**) - Maximum Transmission Unit. The optimal value is the MTU of the interface the tunnel is working over decreased by 20 (so, for 1500-byte ethernet link, set the MTU to 1480 to avoid fragmentation of packets)

mru (*integer*; default: **1480**) - Maximum Receive Unit. The optimal value is the MTU of the interface the tunnel is working over decreased by 20 (so, for 1500-byte ethernet link, set the MTU to 1480 to avoid fragmentation of packets)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

user (*text*; default: **""**) - a user name that is present on the PPPoE server

password (*text*; default: **""**) - a user password used to connect the PPPoE server

profile (*name*) - default profile for the connection

allow (*multiple choice: mschap2*, *mschap1*, *chap*, *pap*; default: **mschap2, mschap1, chap, pap**) - the protocol to allow the client to use for authentication

service-name (*text*; default: **""**) - specifies the service name set on the access concentrator. Leave it blank unless you have many services and need to specify the one you need to connect to

ac-name (*text*; default: **""**) - this may be left blank and the client will connect to any access concentrator that offers the "service" name selected

add-default-route (*yes* | *no*; default: **no**) - whether to add a default route automatically

dial-on-demand (*yes* | *no*; default: **no**) - connects to AC only when outbound traffic is generated and disconnects when there is no traffic for the period set in the idle-timeout value

use-peer-dns (*yes* | *no*; default: **no**) - whether to set the router's default DNS to the PPP peer DNS (i.e. whether to get DNS settings from the peer)

Notes

If there is a default route, **add-default-route** will create a new default route, but will not set it active. In order to make the default route created by add-default-route active all other default routes have to be disabled.

Example

To add and enable PPPoE client on the **gig** interface connecting to the AC that provides **testSN** service using user name **john** with the password **password**:

```
[admin@RemoteOffice] interface pppoe-client> add interface=gig \
\... service-name=testSN user=john password=password disabled=no
[admin@RemoteOffice] interface pppoe-client> print
Flags: X - disabled, R - running
  0  R name="pppoe-out1" mtu=1480 mru=1480 interface=gig user="john"
       password="password" profile=default service-name="testSN" ac-name=""
       add-default-route=no dial-on-demand=no use-peer-dns=no
```

Monitoring PPPoE Client

Command name: ***/interface pppoe-client monitor***

Property Description

status (*text*) - status of the client

- **Dialing** - attempting to make a connection
- **Verifying password...** - connection has been established to the server, password verification in progress
- **Connected** - self-explanatory
- **Terminated** - interface is not enabled or the other side will not establish a connection uptime (time) - connection time displayed in days, hours, minutes and seconds

encoding (*text*) - encryption and encoding (if asymmetric, separated with '/') being used in this

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

connection

uptime (*time*) - connection time displayed in days, hours, minutes and seconds

service-name (*text*) - name of the service the client is connected to

ac-name (*text*) - name of the AC the client is connected to

ac-mac (*MAC address*) - MAC address of the access concentrator (AC) the client is connected to

Example

To monitor the **pppoe-out1** connection:

```
[admin@MikroTik] interface pppoe-client> monitor pppoe-out1
          status: "connected"
          uptime: 10s
        encoding: "none"
    service-name: "testSN"
         ac-name: "10.0.0.1"
          ac-mac: 00:C0:DF:07:5E:E6

[admin@MikroTik] interface pppoe-client>
```

PPPoE Server Setup (Access Concentrator)

Home menu level: ***/interface pppoe-server server***

Description

The PPPoE server (access concentrator) supports multiple servers for each interface - with differing service names. Currently the throughput of the PPPoE server has been tested to 160 Mb/s on a Celeron 600 CPU. Using higher speed CPUs, throughput should increase proportionately.

The **access concentrator name** and PPPoE **service name** are used by clients to identity the access concentrator to register with. The **access concentrator name** is the same as the **identity** of the router displayed before the command prompt. The identity may be set within the **/system identity** submenu.

PPPoE users are created in **/ppp secret** menu, see the ***AAA*** manual for further information.

Note that if no service name is specified in WindowsXP, it will use only service with no name. So if you want to serve WindowsXP clients, leave your service name empty.

Property Description

service-name (*text*) - the PPPoE service name

max-mtu (*integer*; default: **1480**) - Maximum Transmission Unit. The optimal value is the MTU of the interface the tunnel is working over decreased by 20 (so, for 1500-byte Ethernet link, set the MTU to 1480 to avoid fragmentation of packets)

max-mru (*integer*; default: **1480**) - Maximum Receive Unit. The optimal value is the MTU of the interface the tunnel is working over decreased by 20 (so, for 1500-byte Ethernet link, set the MTU to 1480 to avoid fragmentation of packets)

max-sessions (*integer*; default: **0**) - maximum number of clients that the AC can serve

- **0** - unlimited

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

interface (*name*) - interface to which the clients will connect to

authentication (*multiple choice: mschap2 | mschap1 | chap | pap*; default: **mschap2, mschap1, chap, pap**) - authentication algorithm

keepalive-timeout (*time*; default: **10**) - defines the time period (in seconds) after which the router is starting to send keepalive packets every second. If no traffic and no keepalive responses has came for that period of time (i.e. 2 * keepalive-timeout), not responding client is proclaimed disconnected.

one-session-per-host (*yes | no*; default: **no**) - allow only one session per host (determined by MAC address). If a host will try to establish a new session, the old one will be closed

default-profile (*name*; default: **default**) - default profile to use

Notes

The default **keepalive-timeout** value of **10** is OK in most cases. If you set it to **0**, the router will not disconnect clients until they log out or router is restarted. To resolve this problem, the **one-session-per-host** property can be used.

Security issue: do not assign an IP address to the interface you will be receiving the PPPoE requests on.

Example

To add PPPoE server on **ether1** interface providing **ex** service and allowing only one connection per host:

```
[admin@MikroTik] interface pppoe-server server> add interface=ether1 \
\... service-name=ex one-session-per-host=yes
[admin@MikroTik] interface pppoe-server server> print
Flags: X - disabled
  0 X service-name="ex" interface=ether1 mtu=1480 mru=1480
      authentication=mschap2,mschap,chap,pap keepalive-timeout=10
      one-session-per-host=yes default-profile=default

[admin@MikroTik] interface pppoe-server server>
```

PPPoE Server Users

Home menu level: ***/interface pppoe-server***

Property Description

name (*name*) - interface name

service-name (*name*) - name of the service the user is connected to

remote-address (*read-only: MAC address*) - MAC address of the connected client

user (*name*) - the name of the connected user

encoding (*read-only: text*) - encryption and encoding (if asymmetric, separated with '/') being used in this connection

uptime (*time*) - shows how long the client is connected

Example

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

To view the currently connected users:

```
[admin@MikroTik] interface pppoe-server> print
Flags: R - running
  #   NAME       SERVICE REMOTE-ADDRESS    USER    ENCO... UPTIME
  0 R <pppoe-ex> ex      00:C0:CA:16:16:A5 ex              12s

[admin@MikroTik] interface pppoe-server>
```

To disconnect the user **ex**:

```
[admin@MikroTik] interface pppoe-server> remove [find user=ex]
[admin@MikroTik] interface pppoe-server> print

[admin@MikroTik] interface pppoe-server>
```

Application Examples

PPPoE in a multipoint wireless 802.11g network

In a wireless network, the PPPoE server may be attached to an Access Point (as well as to a regular station of wireless infrastructure). Either our RouterOS client or Windows PPPoE clients may connect to the Access Point for PPPoE authentication. Further, for RouterOS clients, the radio interface may be set to MTU 1600 so that the PPPoE interface may be set to MTU 1500. This optimizes the transmission of 1500 byte packets and avoids any problems associated with MTUs lower than 1500. It has not been determined how to change the MTU of the Windows wireless interface at this moment.

Let us consider the following setup where the MikroTik Wireless AP offers wireless clients transparent access to the local network with authentication:

First of all, the wireless interface should be configured:

```
[admin@PPPoE-Server] interface wireless> set 0 mode=ap-bridge \
   frequency=2442 band=2.4ghz-b/g ssid=mt disabled=no
[admin@PPPoE-Server] interface wireless> print
Flags: X - disabled, R - running
 0    name="wlan1" mtu=1500 mac-address=00:01:24:70:53:04 arp=enabled
      disable-running-check=no interface-type=Atheros AR5211
      radio-name="000124705304" mode=station ssid="mt" area=""
      frequency-mode=superchannel country=no_country_set antenna-gain=0
      frequency=2412 band=2.4ghz-b scan-list=default rate-set=default
      supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
      supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,
                          54Mbps
      basic-rates-b=1Mbps basic-rates-a/g=6Mbps max-station-count=2007
      ack-timeout=dynamic tx-power=default tx-power-mode=default
      noise-floor-threshold=default periodic-calibration=default
      burst-time=disabled fast-frames=no dfs-mode=none antenna-mode=ant-a
      wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
      update-stats-interval=disabled default-authentication=yes
      default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
      hide-ssid=no security-profile=default disconnect-timeout=3s
      on-fail-retry-time=100ms preamble-mode=both
[admin@PPPoE-Server] interface wireless>
```

Now, configure the Ethernet interface, add the IP address and set the default route:

```
[admin@PPPoE-Server] ip address> add address=10.1.0.3/24 interface=Local
[admin@PPPoE-Server] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   10.1.0.3/24        10.1.0.0        10.1.0.255      Local
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@PPPoE-Server] ip address> /ip route
[admin@PPPoE-Server] ip route> add gateway=10.1.0.1
[admin@PPPoE-Server] ip route> print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf
 #     DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
 0 ADC 10.1.0.0/24                                 Local
 1 A S 0.0.0.0/0          r 10.1.0.1        1      Local
[admin@PPPoE-Server] ip route> /interface ethernet
[admin@PPPoE-Server] interface ethernet> set Local arp=proxy-arp
[admin@PPPoE-Server] interface ethernet> print
Flags: X - disabled, R - running
 #    NAME                                  MTU   MAC-ADDRESS       ARP
 0  R Local                                 1500  00:0C:42:03:25:53 proxy-arp
[admin@PPPoE-Server] interface ethernet>
```

We should add PPPoE server to the wireless interface:

```
[admin@PPPoE-Server] interface pppoe-server server> add interface=wlan1 \
   service-name=mt one-session-per-host=yes disabled=no
[admin@PPPoE-Server] interface pppoe-server server> print
Flags: X - disabled
 0   service-name="mt" interface=wlan1 max-mtu=1480 max-mru=1480
     authentication=pap,chap,mschap1,mschap2 keepalive-timeout=10
     one-session-per-host=yes max-sessions=0 default-profile=default
[admin@PPPoE-Server] interface pppoe-server server>
```

Finally, we can set up PPPoE clients:

```
[admin@PPPoE-Server] ip pool> add name=pppoe ranges=10.1.0.100-10.1.0.200
[admin@PPPoE-Server] ip pool> print
 # NAME                                      RANGES
 0 pppoe                                     10.1.0.100-10.1.0.200
[admin@PPPoE-Server] ip pool> /ppp profile
[admin@PPPoE-Server] ppp profile> set default use-encryption=yes \
   local-address=10.1.0.3 remote-address=pppoe
[admin@PPPoE-Server] ppp profile> print
Flags: * - default
 0 * name="default" local-address=10.1.0.3 remote-address=pppoe
     use-compression=no use-vj-compression=no use-encryption=yes only-one=no
     change-tcp-mss=yes

 1 * name="default-encryption" use-compression=default
     use-vj-compression=default use-encryption=yes only-one=default
     change-tcp-mss=default
[admin@PPPoE-Server] ppp profile> .. secret
[admin@PPPoE-Server] ppp secret> add name=w password=wkst service=pppoe
[admin@PPPoE-Server] ppp secret> add name=l password=ltp service=pppoe
[admin@PPPoE-Server] ppp secret> print
Flags: X - disabled
 #   NAME        SERVICE CALLER-ID PASSWORD  PROFILE           REMOTE-ADDRESS
 0   w           pppoe             wkst      default           0.0.0.0
 1   l           pppoe             ltp       default           0.0.0.0
[admin@PPPoE-Server] ppp secret>
```

Thus we have completed the configuration and added two users: **w** and **l** who are able to connect to Internet, using PPPoE client software.

Note that Windows XP built-in client supports encryption, but RASPPPOE does not. So, if it is planned not to support Windows clients older than Windows XP, it is recommended to switch **require-encryption** to **yes** value in the **default** profile configuration. In other case, the server will accept clients that do not encrypt data.

Troubleshooting

Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **I can connect to my PPPoE server. The ping goes even through it, but I still cannot open web pages**
  Make sure that you have specified a valid DNS server in the router (in **/ip dns** or in **/ppp profile** the **dns-server** parameter).
- **The PPPoE server shows more than one active user entry for one client, when the clients disconnect, they are still shown and active**
  Set the **keepalive-timeout** parameter (in the PPPoE server configuration) to **10** if You want clients to be considered logged off if they do not respond for 10 seconds.
  Note that if the **keepalive-timeout** parameter is set to **0** and the **only-one** parameter (in PPP profile settings) is set to **yes** then the clients might be able to connect only once. To resolve this problem **one-session-per-host** parameter in PPPoE server configuration should be set to **yes**
- **I can get through the PPPoE link only small packets (eg. pings)**
  You need to change **mss** of all the packets passing through the PPPoE link to the value of PPPoE link's MTU-40 at least on one of the peers. So for PPPoE link with MTU of 1480:

```
[admin@MT] interface pppoe-server server> set 0 max-mtu=1440 max-mru=1440
[admin@MT] interface pppoe-server server> print
Flags: X - disabled
 0   service-name="mt" interface=wlan1 max-mtu=1440 max-mru=1440
     authentication=pap,chap,mschap1,mschap2 keepalive-timeout=10
     one-session-per-host=yes max-sessions=0 default-profile=default
[admin@MT] interface pppoe-server server>
```

- **My windows PPPoE client obtains IP address and default gateway from the MikroTik PPPoE server, but it cannot ping beyond the PPPoE server and use the Internet**
  PPPoE server is not bridging the clients. Configure masquerading for the PPPoE client addresses, or make sure you have proper routing for the address space used by the clients, or you enable Proxy-ARP on the Ethernet interface (See the IP Addresses and Address Resolution Protocol (ARP) Manual)
- **My Windows XP client cannot connect to the PPPoE server**
  You have to specify the "Service Name" in the properties of the XP PPPoE client. If the service name is not set, or it does not match the service name of the MikroTik PPPoE server, you get the "line is busy" errors, or the system shows "verifying password - unknown error"
- **I want to have logs for PPPoE connection establishment**
  Configure the logging feature under the **/system logging facility** and enable the PPP type logs

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

PPTP

Document revision 1.4 (Tue Aug 09 12:01:21 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Quick Setup Guide
  Specifications
  Related Documents
  Description
  Additional Documents
PPTP Client Setup
  Property Description
  Example
Monitoring PPTP Client
  Property Description
  Example
PPTP Server Setup
  Description
  Property Description
  Example
PPTP Server Users
  Description
  Property Description
  Example
PPTP Application Examples
  Router-to-Router Secure Tunnel Example
  Connecting a Remote Client via PPTP Tunnel
  PPTP Setup for Windows
  Sample instructions for PPTP (VPN) installation and client setup - Windows 98SE
Troubleshooting
  Description

General Information

Summary

PPTP (Point to Point Tunnel Protocol) supports encrypted tunnels over IP. The MikroTik RouterOS implementation includes support for PPTP client and server.

General applications of PPTP tunnels:

- For secure router-to-router tunnels over the Internet
- To link (bridge) local Intranets or LANs (when EoIP is also used)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- For mobile or remote clients to remotely access an Intranet/LAN of a company (see PPTP setup for Windows for more information)

Each PPTP connection is composed of a server and a client. The MikroTik RouterOS may function as a server or client - or, for various configurations, it may be the server for some connections and client for other connections. For example, the client created below could connect to a Windows 2000 server, another MikroTik Router, or another router which supports a PPTP server.

Quick Setup Guide

To make a PPTP tunnel between 2 MikroTik routers with IP addresses **10.5.8.104** (PPTP server) and **10.1.0.172** (PPTP client), follow the next steps.

- Setup on PPTP server:
  1. Add a user:

```
[admin@PPTP-Server] ppp secret> add name=jack password=pass \
\... local-address=10.0.0.1 remote-address=10.0.0.2
```

  2. Enable the PPTP server:

```
[admin@PPTP-Server] interface pptp-server server> set enabled=yes
```

- Setup on PPTP client:
  1. Add the PPTP client:

```
[admin@PPTP-Client] interface pptp-client> add user=jack password=pass \
\... connect-to=10.5.8.104 disabled=no
```

Specifications

Packages required: ***ppp***
License required: ***level1 (limited to 1 tunnel), level3 (limited to 200 tunnels), level5***
Home menu level: ***/interface pptp-server, /interface pptp-client***
Standards and Technologies: ***PPTP (RFC 2637)***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***IP Addresses and ARP***
- ***PPP User AAA***
- ***EoIP***

Description

PPTP is a secure tunnel for transporting IP traffic using PPP. PPTP encapsulates PPP in virtual lines that run over IP. PPTP incorporates PPP and MPPE (Microsoft Point to Point Encryption) to make encrypted links. The purpose of this protocol is to make well-managed secure connections between

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

routers as well as between routers and PPTP clients (clients are available for and/or included in almost all OSs including Windows).

PPTP includes PPP authentication and accounting for each PPTP connection. Full authentication and accounting of each connection may be done through a RADIUS client or locally.

MPPE 40bit RC4 and MPPE 128bit RC4 encryption are supported.

PPTP traffic uses TCP port 1723 and IP protocol GRE (Generic Routing Encapsulation, IP protocol ID 47), as assigned by the Internet Assigned Numbers Authority (IANA). PPTP can be used with most firewalls and routers by enabling traffic destined for TCP port 1723 and protocol 47 traffic to be routed through the firewall or router.

PPTP connections may be limited or impossible to setup though a masqueraded/NAT IP connection. Please see the Microsoft and RFC links at the end of this section for more information.

Additional Documents

- ***http://msdn.microsoft.com/library/backgrnd/html/understanding_pptp.htm***
- ***http://support.microsoft.com/support/kb/articles/q162/8/47.asp***
- ***http://www.ietf.org/rfc/rfc2637.txt?number=2637***
- ***http://www.ietf.org/rfc/rfc3078.txt?number=3078***
- ***http://www.ietf.org/rfc/rfc3079.txt?number=3079***

PPTP Client Setup

Home menu level: ***/interface pptp-client***

Property Description

add-default-route (*yes* | *no*; default: **no**) - whether to use the server which this client is connected to as its default router (gateway)

allow (*multiple choice: mschap2, mschap1, chap, pap*; default: **mschap2, mschap1, chap, pap**) - the protocol to allow the client to use for authentication

connect-to (*IP address*) - The IP address of the PPTP server to connect to

mru (*integer*; default: **1460**) - Maximum Receive Unit. The optimal value is the MRU of the interface the tunnel is working over decreased by 40 (so, for 1500-byte ethernet link, set the MRU to 1460 to avoid fragmentation of packets)

mtu (*integer*; default: **1460**) - Maximum Transmission Unit. The optimal value is the MTU of the interface the tunnel is working over decreased by 40 (so, for 1500-byte ethernet link, set the MTU to 1460 to avoid fragmentation of packets)

name (*name*; default: **pptp-outN**) - interface name for reference

password (*text*; default: **""**) - user password to use when logging to the remote server

profile (*name*; default: **default**) - profile to use when connecting to the remote server

user (*text*) - user name to use when logging on to the remote server

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

To set up PPTP client named **test2** using username **john** with password **john** to connect to the **10.1.1.12** PPTP server and use it as the default gateway:

```
[admin@MikroTik] interface pptp-client> add name=test2 connect-to=10.1.1.12 \
\... user=john add-default-route=yes password=john
[admin@MikroTik] interface pptp-client> print
Flags: X - disabled, R - running
  0 X  name="test2" mtu=1460 mru=1460 connect-to=10.1.1.12 user="john"
       password="john" profile=default add-default-route=yes


[admin@MikroTik] interface pptp-client> enable 0
```

Monitoring PPTP Client

Command name: ***/interface pptp-client monitor***

Property Description

encoding (*text*) - encryption and encoding (if asymmetric, seperated with '/') being used in this connection

status (*text*) - status of the client

- **Dialing** - attempting to make a connection
- **Verifying password...** - connection has been established to the server, password verification in progress
- **Connected** - self-explanatory
- **Terminated** - interface is not enabled or the other side will not establish a connection uptime (time) - connection time displayed in days, hours, minutes and seconds

uptime (*time*) - connection time displayed in days, hours, minutes and seconds

Example

Example of an established connection:

```
[admin@MikroTik] interface pptp-client> monitor test2
      uptime: 4h35s
    encoding: MPPE 128 bit, stateless
      status: Connected
[admin@MikroTik] interface pptp-client>
```

PPTP Server Setup

Home menu level: ***/interface pptp-server server***

Description

The PPTP server creates a dynamic interface for each connected PPTP client. The PPTP connection count from clients depends on the license level you have. Level1 license allows 1 PPTP client, Level3 or Level4 licenses up to 200 clients, and Level5 or Level6 licenses do not have PPTP client limitations.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

To create PPTP users, you should consult the ***PPP secret*** and ***PPP Profile*** manuals. It is also possible to use the MikroTik router as a RADIUS client to register the PPTP users, see the ***manual*** how to do it.

Property Description

authentication (*multiple choice: pap | chap | mschap1 | mschap2*; default: **mschap2**) - authentication algorithm

default-profile - default profile to use

enabled (*yes | no*; default: **no**) - defines whether PPTP server is enabled or not

keepalive-timeout (*time*; default: **30**) - defines the time period (in seconds) after which the router is starting to send keepalive packets every second. If no traffic and no keepalive responses has came for that period of time (i.e. 2 * keepalive-timeout), not responding client is proclaimed disconnected

mru (*integer*; default: **1460**) - Maximum Receive Unit. The optimal value is the MRU of the interface the tunnel is working over decreased by 40 (so, for 1500-byte ethernet link, set the MRU to 1460 to avoid fragmentation of packets)

mtu (*integer*; default: **1460**) - Maximum Transmission Unit. The optimal value is the MTU of the interface the tunnel is working over decreased by 40 (so, for 1500-byte ethernet link, set the MTU to 1460 to avoid fragmentation of packets)

Example

To enable PPTP server:

```
[admin@MikroTik] interface pptp-server server> set enabled=yes
[admin@MikroTik] interface pptp-server server> print
               enabled: yes
                   mtu: 1460
                   mru: 1460
        authentication: mschap2,mschap1
    keepalive-timeout: 30
      default-profile: default
[admin@MikroTik] interface pptp-server server>
```

PPTP Server Users

Home menu level: ***/interface pptp-server***

Description

There are two types of items in PPTP server configuration - static users and dynamic connections. A dynamic connection can be established if the user database or the **default-profile** has its **local-address** and **remote-address** set correctly. When static users are added, the default profile may be left with its default values and only PPP user (in **/ppp secret**) should be configured. **Note** that in both cases PPP users must be configured properly.

Property Description

client-address (*IP address*) - shows (cannot be set here) the IP address of the connected client

encoding (*text*) - encryption and encoding (if asymmetric, separated with '/') being used in this

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

connection

mtu (*integer*) - (cannot be set here) client's MTU

name (*name*) - interface name

uptime (*time*) - shows how long the client is connected

user (*name*) - the name of the user that is configured statically or added dynamically

Example

To add a static entry for **ex1** user:

```
[admin@MikroTik] interface pptp-server> add user=ex1
[admin@MikroTik] interface pptp-server> print
Flags: X - disabled, D - dynamic, R - running
  #     NAME                USER          MTU   CLIENT-ADDRESS  UPTIME   ENC...
  0  DR <pptp-ex>           ex            1460  10.0.0.202      6m32s    none
  1     pptp-in1            ex1
[admin@MikroTik] interface pptp-server>
```

In this example an already connected user **ex** is shown besides the one we just added.

PPTP Application Examples

Router-to-Router Secure Tunnel Example

The following is an example of connecting two Intranets using an encrypted PPTP tunnel over the Internet.

There are two routers in this example:

- [HomeOffice]
  Interface LocalHomeOffice 10.150.2.254/24
  Interface ToInternet 192.168.80.1/24

- [RemoteOffice]
  Interface ToInternet 192.168.81.1/24
  Interface LocalRemoteOffice 10.150.1.254/24

Each router is connected to a different ISP. One router can access another router through the Internet.

On the `Preforma` PPTP server a user must be set up for the client:

```
[admin@HomeOffice] ppp secret> add name=ex service=pptp password=lkjrht
local-address=10.0.103.1 remote-address=10.0.103.2
[admin@HomeOffice] ppp secret> print detail
Flags: X - disabled
  0   name="ex" service=pptp caller-id="" password="lkjrht" profile=default
      local-address=10.0.103.1 remote-address=10.0.103.2 routes==""

[admin@HomeOffice] ppp secret>
```

Then the user should be added in the PPTP server list:

```
[admin@HomeOffice] interface pptp-server> add user=ex
[admin@HomeOffice] interface pptp-server> print
Flags: X - disabled, D - dynamic, R - running
  #     NAME                USER          MTU   CLIENT-ADDRESS  UPTIME   ENC...
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  0     pptp-in1             ex
[admin@HomeOffice] interface pptp-server>
```

And finally, the server must be enabled:

```
[admin@HomeOffice] interface pptp-server server> set enabled=yes
[admin@HomeOffice] interface pptp-server server> print
            enabled: yes
                mtu: 1460
                mru: 1460
     authentication: mschap2
    default-profile: default
[admin@HomeOffice] interface pptp-server server>
```

Add a PPTP client to the RemoteOffice router:

```
[admin@RemoteOffice] interface pptp-client> add connect-to=192.168.80.1 user=ex \
\... password=lkjrht disabled=no
[admin@RemoteOffice] interface pptp-client> print
Flags: X - disabled, R - running
  0  R name="pptp-out1" mtu=1460 mru=1460 connect-to=192.168.80.1 user="ex"
       password="lkjrht" profile=default add-default-route=no


[admin@RemoteOffice] interface pptp-client>
```

Thus, a PPTP tunnel is created between the routers. This tunnel is like an Ethernet point-to-point connection between the routers with IP addresses 10.0.103.1 and 10.0.103.2 at each router. It enables 'direct' communication between the routers over third party networks.

To route the local Intranets over the PPTP tunnel you need to add these routes:

```
[admin@HomeOffice] > ip route add dst-address 10.150.1.0/24 gateway 10.0.103.2
[admin@RemoteOffice] > ip route add dst-address 10.150.2.0/24 gateway 10.0.103.1
```

On the PPTP server it can alternatively be done using **routes** parameter of the user configuration:

```
[admin@HomeOffice] ppp secret> print detail
Flags: X - disabled
  0   name="ex" service=pptp caller-id="" password="lkjrht" profile=default
      local-address=10.0.103.1 remote-address=10.0.103.2 routes==""

[admin@HomeOffice] ppp secret> set 0 routes="10.150.1.0/24 10.0.103.2 1"
[admin@HomeOffice] ppp secret> print detail
Flags: X - disabled
  0   name="ex" service=pptp caller-id="" password="lkjrht" profile=default
      local-address=10.0.103.1 remote-address=10.0.103.2
      routes="10.150.1.0/24 10.0.103.2 1"

[admin@HomeOffice] ppp secret>
```

Test the PPTP tunnel connection:

```
[admin@RemoteOffice]> /ping 10.0.103.1
10.0.103.1 pong: ttl=255 time=3 ms
10.0.103.1 pong: ttl=255 time=3 ms
10.0.103.1 pong: ttl=255 time=3 ms
ping interrupted
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 3/3.0/3 ms
```

Test the connection through the PPTP tunnel to the LocalHomeOffice interface:

```
[admin@RemoteOffice]> /ping 10.150.2.254
10.150.2.254 pong: ttl=255 time=3 ms
10.150.2.254 pong: ttl=255 time=3 ms
10.150.2.254 pong: ttl=255 time=3 ms
ping interrupted
3 packets transmitted, 3 packets received, 0% packet loss
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
round-trip min/avg/max = 3/3.0/3 ms
```

To bridge a LAN over this secure tunnel, please see the example in the 'EoIP' section of the manual. To set the maximum speed for traffic over this tunnel, please consult the 'Queues' section.

Connecting a Remote Client via PPTP Tunnel

The following example shows how to connect a computer to a remote office network over PPTP encrypted tunnel giving that computer an IP address from the same network as the remote office has (without need of bridging over EoIP tunnels)

Please, consult the respective manual on how to set up a PPTP client with the software You are using.

The router in this example:

- [RemoteOffice]
  Interface ToInternet 192.168.81.1/24
  Interface Office 10.150.1.254/24

The client computer can access the router through the Internet.

On the PPTP server a user must be set up for the client:

```
[admin@RemoteOffice] ppp secret> add name=ex service=pptp password=lkjrht
local-address=10.150.1.254 remote-address=10.150.1.2
[admin@RemoteOffice] ppp secret> print detail
Flags: X - disabled
  0   name="ex" service=pptp caller-id="" password="lkjrht" profile=default
      local-address=10.150.1.254 remote-address=10.150.1.2 routes==""

[admin@RemoteOffice] ppp secret>
```

Then the user should be added in the PPTP server list:

```
[admin@RemoteOffice] interface pptp-server> add name=FromLaptop user=ex
[admin@RemoteOffice] interface pptp-server> print
Flags: X - disabled, D - dynamic, R - running
  #     NAME                USER         MTU   CLIENT-ADDRESS  UPTIME   ENC...
  0     FromLaptop          ex
[admin@RemoteOffice] interface pptp-server>
```

And the server must be enabled:

```
[admin@RemoteOffice] interface pptp-server server> set enabled=yes
[admin@RemoteOffice] interface pptp-server server> print
             enabled: yes
                 mtu: 1460
                 mru: 1460
      authentication: mschap2
    default-profile: default
[admin@RemoteOffice] interface pptp-server server>
```

Finally, the proxy APR must be enabled on the 'Office' interface:

```
[admin@RemoteOffice] interface ethernet> set Office arp=proxy-arp
[admin@RemoteOffice] interface ethernet> print
Flags: X - disabled, R - running
  #    NAME                MTU   MAC-ADDRESS       ARP
  0  R ToInternet          1500  00:30:4F:0B:7B:C1 enabled
  1  R Office              1500  00:30:4F:06:62:12 proxy-arp
[admin@RemoteOffice] interface ethernet>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

PPTP Setup for Windows

Microsoft provides PPTP client support for Windows NT, 2000, ME, 98SE, and 98. Windows 98SE, 2000, and ME include support in the Windows setup or automatically install PPTP. For 95, NT, and 98, installation requires a download from Microsoft. Many ISPs have made help pages to assist clients with Windows PPTP installation.

- ***http://www.real-time.com/Customer_Support/PPTP_Config/pptp_config.html***
- ***http://www.microsoft.com/windows95/downloads/contents/WUAdminTools/S_WUNetworkingTools/W95Wi***

Sample instructions for PPTP (VPN) installation and client setup - Windows 98SE

If the VPN (PPTP) support is installed, select 'Dial-up Networking' and 'Create a new connection'. The option to create a 'VPN' should be selected. If there is no 'VPN' options, then follow the installation instructions below. When asked for the 'Host name or IP address of the VPN server', type the IP address of the router. Double-click on the 'new' icon and type the correct user name and password (must also be in the user database on the router or RADIUS server used for authentication).

The setup of the connections takes nine seconds after selection the 'connect' button. It is suggested that the connection properties be edited so that 'NetBEUI', 'IPX/SPX compatible', and 'Log on to network' are unselected. The setup time for the connection will then be two seconds after the 'connect' button is selected.

To install the 'Virtual Private Networking' support for Windows 98SE, go to the 'Setting' menu from the main 'Start' menu. Select 'Control Panel', select 'Add/Remove Program', select the 'Windows setup' tab, select the 'Communications' software for installation and 'Details'. Go to the bottom of the list of software and select 'Virtual Private Networking' to be installed.

Troubleshooting

Description

- **I use firewall and I cannot establish PPTP connection**
  Make sure the TCP connections to port 1723 can pass through both directions between your sites. Also, IP protocol 47 should be passed through

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

VLAN

Document revision 1.2 (Mon Sep 19 13:46:34 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
  Additional Documents
VLAN Setup
  Property Description
  Notes
  Example
Application Example
  VLAN example on MikroTik Routers

General Information

Summary

VLAN is an implementation of the 802.1Q VLAN protocol for MikroTik RouterOS. It allows you to have multiple Virtual LANs on a single ethernet or wireless interface, giving the ability to segregate LANs efficiently. It supports up to 4095 vlan interfaces, each with a unique VLAN ID, per ethernet device. Many routers, including Cisco and Linux based, and many Layer 2 switches also support it.

A VLAN is a logical grouping that allows end users to communicate as if they were physically connected to a single isolated LAN, independent of the physical configuration of the network. VLAN support adds a new dimension of security and cost savings permitting the sharing of a physical network while logically maintaining separation among unrelated users.

Specifications

Packages required: ***system***
License required: ***level1 (limited to 1 vlan), level3***
Home menu level: ***/interface vlan***
Standards and Technologies: ***VLAN (IEEE 802.1Q)***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***IP Addresses and ARP***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

VLANs are simply a way of grouping a set of switch ports together so that they form a logical network, separate from any other such group. Within a single switch this is straightforward local configuration. When the VLAN extends over more than one switch, the inter-switch links have to become trunks, on which packets are tagged to indicate which VLAN they belong to.

You can use MikroTik RouterOS (as well as Cisco IOS and Linux) to mark these packets as well as to accept and route marked ones.

As VLAN works on OSI Layer 2, it can be used just as any other network interface without any restrictions. And VLAN successfully passes through Ethernet bridges (for MikroTik RouterOS bridges you should set **forward-protocols** to **ip**, **arp** and **other**; for other bridges there should be analogical settings).

You can also transport VLANs over wireless links and put multiple VLAN interfaces on a single wireless interface. Note that as VLAN is not a full tunnel protocol (i.e., it does not have additional fields to transport MAC addresses of sender and recipient), the same limitation applies to bridging over VLAN as to bridging plain wireless interfaces. In other words, while wireless clients may participate in VLANs put on wireless interfaces, it is not possible to have VLAN put on a wireless interface in station mode bridged with any other interface.

Currently supported Ethernet interfaces

This is a list of network interfaces on which VLAN was tested and worked. Note that there might be many other interfaces that support VLAN, but they just were not checked.

- Realtek 8139
- Intel PRO/100
- Intel PRO1000 server adapter
- National Semiconductor DP83816 based cards (RouterBOARD200 onboard Ethernet, RouterBOARD 24 card)
- National Semiconductor DP83815 (Soekris onboard Ethernet)
- VIA VT6105M based cards (RouterBOARD 44 card)
- VIA VT6105
- VIA VT6102 (VIA EPIA onboard Ethernet)

This is a list of network interfaces on which VLAN was tested and worked, but **WITHOUT LARGE PACKET (>1496 bytes) SUPPORT**:

- 3Com 3c59x PCI
- DEC 21140 (tulip)

Additional Documents

- ***http://www.csd.uwo.ca/courses/CS457a/reports/handin/jpbojtos/A2/trunking.htm***
- ***http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121t/121t3/dtbridge.htm#xtocid***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***http://www.cisco.com/warp/public/473/27.html#tagging***
- ***http://www.cisco.com/warp/public/538/7.html***
- ***http://www.nwfusion.com/news/tech/2001/0305tech.html***
- ***http://www.intel.com/network/connectivity/resources/doc_library/tech_brief/virtual_lans.htm***

VLAN Setup

Home menu level: ***/interface vlan***

Property Description

arp (*disabled* | *enabled* | *proxy-arp* | *reply-only*; default: **enabled**) - Address Resolution Protocol setting

- **disabled** - the interface will not use ARP protocol
- **enabled** - the interface will use ARP protocol
- **proxy-arp** - the interface will be an ARP proxy
- **reply-only** - the interface will only reply to the requests originated to its own IP addresses, but neighbor MAC addresses will be gathered from /ip arp statically set table only

interface (*name*) - physical interface to the network where are VLANs

mtu (*integer*; default: **1500**) - Maximum Transmission Unit

name (*name*) - interface name for reference

vlan-id (*integer*; default: **1**) - Virtual LAN identifier or tag that is used to distinguish VLANs. Must be equal for all computers in one VLAN.

Notes

MTU should be set to 1500 bytes as on Ethernet interfaces. But this may not work with some Ethernet cards that do not support receiving/transmitting of full size Ethernet packets with VLAN header added (1500 bytes data + 4 bytes VLAN header + 14 bytes Ethernet header). In this situation MTU 1496 can be used, but note that this will cause packet fragmentation if larger packets have to be sent over interface. At the same time remember that MTU 1496 may cause problems if path MTU discovery is not working properly between source and destination.

Example

To add and enable a VLAN interface named **test** with **vlan-id**=1 on interface **ether1**:

```
[admin@MikroTik] interface vlan> add name=test vlan-id=1 interface=ether1
[admin@MikroTik] interface vlan> print
Flags: X - disabled, R - running
  #    NAME                MTU  ARP       VLAN-ID INTERFACE
  0 X  test                1500 enabled   1       ether1
[admin@MikroTik] interface vlan> enable 0
[admin@MikroTik] interface vlan> print
Flags: X - disabled, R - running
  #    NAME                MTU  ARP       VLAN-ID INTERFACE
  0  R test                1500 enabled   1       ether1
[admin@MikroTik] interface vlan>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Application Example

VLAN example on MikroTik Routers

Let us assume that we have two or more MikroTik RouterOS routers connected with a hub. Interfaces to the physical network, where VLAN is to be created is **ether1** for all of them (it is needed only for example simplification, it is NOT a must).

To connect computers through VLAN they must be connected physically and unique IP addresses should be assigned them so that they could ping each other. Then on each of them the VLAN interface should be created:

```
[admin@MikroTik] interface vlan> add name=test vlan-id=32 interface=ether1
[admin@MikroTik] interface vlan> print
Flags: X - disabled, R - running
  #    NAME                MTU  ARP       VLAN-ID INTERFACE
  0  R test                1500 enabled   32      ether1
[admin@MikroTik] interface vlan>
```

If the interfaces were successfully created, both of them will be **running**. If computers are connected incorrectly (through network device that does not retransmit or forward VLAN packets), either both or one of the interfaces will not be **running**.

When the interface is running, IP addresses can be assigned to the VLAN interfaces.

On the Router 1:

```
[admin@MikroTik] ip address> add address=10.10.10.1/24 interface=test
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.204/24      10.0.0.0        10.0.0.255      ether1
  1   10.20.0.1/24       10.20.0.0       10.20.0.255     pc1
  2   10.10.10.1/24      10.10.10.0      10.10.10.255    test
[admin@MikroTik] ip address>
```

On the Router 2:

```
[admin@MikroTik] ip address> add address=10.10.10.2/24 interface=test
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.201/24      10.0.0.0        10.0.0.255      ether1
  1   10.10.10.2/24      10.10.10.0      10.10.10.255    test
[admin@MikroTik] ip address>
```

If it set up correctly, then it is possible to ping Router 2 from Router 1 and vice versa:

```
[admin@MikroTik] ip address> /ping 10.10.10.1
10.10.10.1 64 byte pong: ttl=255 time=3 ms
10.10.10.1 64 byte pong: ttl=255 time=4 ms
10.10.10.1 64 byte pong: ttl=255 time=10 ms
10.10.10.1 64 byte pong: ttl=255 time=5 ms
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 3/10.5/10 ms
[admin@MikroTik] ip address> /ping 10.10.10.2
10.10.10.2 64 byte pong: ttl=255 time=10 ms
10.10.10.2 64 byte pong: ttl=255 time=11 ms
10.10.10.2 64 byte pong: ttl=255 time=10 ms
10.10.10.2 64 byte pong: ttl=255 time=13 ms
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 10/11/13 ms
[admin@MikroTik] ip address>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Graphing

Document revision 1.0 (09-08-2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Description
General Options
  Property Description
  Example
Health Graphing
  Description
  Property Description
Interface Graphing
  Description
  Property Description
  Example
Simple Queue Graphing
  Description
  Property Description
  Example
Resource Graphing
  Description
  Property Description
  Example

General Information

Summary

Graphing is a tool which is used for monitoring various RouterOS parameters over a period of time.

Specifications

Packages required: ***system, routerboard (optional)***
License required: ***level1***
Home menu level: ***/tool graphing***
Hardware usage: ***Not significant***

Description

The Graphing tool can display graphics for:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- Routerboard health (voltage and temperature)
- Resource usage (CPU, Memory and Disk usage)
- Traffic which is passed through interfaces
- Traffic which is passed through simple queues

Graphing consists of two parts - first part collects information and other part displays data in a Web page. To access the graphics, type **http://[Router_IP_address]/graphs/** and choose a graphic to display in your Web browser.

Data from the router is gathered every 5 minutes, but saved on the system drive every **store-every** time. After rebooting the router, graphing will display information that was last time saved on the disk before the reboot.

RouterOS generates four graphics for each item:

- "Daily" Graph (5 Minute Average)
- "Weekly" Graph (30 Minute Average)
- "Monthly" Graph (2 Hour Average)
- "Yearly" Graph (1 Day Average)

To access each graphic from a network, specify this network in **allow-address** parameter for the respective item.

General Options

Home menu level: ***/tool graphing***

Property Description

store-every (*5min* | *hour* | *24hours*; default: **5min**) - how often to store information on system drive

Example

To store information on system drive every hour:

```
/tool graphing set store-every=hour
[admin@MikroTik] tool graphing> print
    store-every: hour
[admin@MikroTik] tool graphing>
```

Health Graphing

Home menu level: ***/tool graphing health***

Description

This submenu provides information about RouterBoard's 'health' - voltage and temperature. For this option, you have to install the **routerboard** package:

Property Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

allow-address (*IP address* | *netmask*; default: **0.0.0.0/0**) - network which is allowed to view graphs of router health

store-on-disk (yes | no; default: **yes**) - whether to store information about traffic on system drive or not. If not, the information will be stored in RAM and will be lost after a reboot

Interface Graphing

Home menu level: ***/tool graphing interface***

Description

Shows how much traffic is passed through an interface over a period of time.

Property Description

allow-address (*IP address* | *netmask*; default: **0.0.0.0/0**) - IP address range which is allowed to view information about the interface. If a client PC not belonging to this IP address range tries to open http://[Router_IP_address]/graphs/, it will not see this entry

interface (*name*; default: **all**) - name of the interface which will be monitored

store-on-disk (yes | no; default: **yes**) - whether to store information about traffic on system drive or not. If not, the information will be stored in RAM and will be lost after a reboot

Example

To monitor traffic which is passed through interface **ether1** only from local network **192.168.0.0/24**, and write information on disk:

```
[admin@MikroTik] tool graphing interface> add interface=ether1 \
\... allow-address=192.168.0.0/24 store-on-disk=yes
[admin@MikroTik] tool graphing interface> print
Flags: X - disabled
 #   INTERFACE ALLOW-ADDRESS      STORE-ON-DISK
 0   ether1    192.168.0.0/24     yes
[admin@MikroTik] tool graphing interface>
```

Graph for interface **ether1**:

Simple Queue Graphing

Home menu level: ***/tool graphing queue***

Description

In this submenu you can specify a queue from the **/queue simple** list to make a graphic for it.

Property Description

allow-address (*IP address* | *netmask*; default: **0.0.0.0/0**) - IP address range which is allowed to view information about the queue. If a client PC not belonging to this IP address range tries to open http://[Router_IP_address]/graphs/, it will not see this entry

allow-target (yes | no; default: **yes**) - whether to allow access to web graphing from IP range that is

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

specified in /queue simple target-address

simple-queue (*name*; default: **all**) - name of simple queue which will be monitored

store-on-disk (yes | no; default: **yes**) - whether to store information about traffic on hard drive or not. If not, the information will be stored in RAM and will be lost after a reboot

Example

Add a simple queue to Grapher list with simple-queue name **queue1**, allow limited clients to access Grapher from web, store information about traffic on disk:

```
[admin@MikroTik] tool graphing queue> add simple-queue=queue1 allow-address=yes \
\... store-on-disk=yes
```

"Daily" graphic for **queue1**:

Resource Graphing

Home menu level: ***/tool graphing resource***

Description

Provides with router resource usage information over a period of time:

- CPU usage
- Memory usage
- Disk usage

Property Description

allow-address (*IP address* | *netmask*; default: **0.0.0.0/0**) - IP address range which is allowed to view information about the resource usage. If a client PC not belonging to this IP address range tries to open http://[Router_IP_address]/graphs/, it will not see this entry

store-on-disk (yes | no; default: **yes**) - whether to store information about traffic on hard drive or not. If not, the information will be stored in RAM and will be lost after a reboot

Example

Add IP range **192.168.0.0/24** from which users are allowed to monitor Grapher's resource usage:

```
[admin@MikroTik] tool graphing resource> add allow-address=192.168.0.0/24 \
\... store-on-disk=yes
[admin@MikroTik] tool graphing resource> print
Flags: X - disabled
 #   ALLOW-ADDRESS      STORE-ON-DISK
 0   192.168.0.0/24     yes
[admin@MikroTik] tool graphing resource>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

HotSpot User AAA

Document revision 2.3 (Tue Sep 27 14:30:17 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
HotSpot User Profiles
  Description
  Property Description
  Notes
  Example
HotSpot Users
  Property Description
  Notes
  Example
HotSpot Active Users
  Description
  Property Description
  Example

General Information

Summary

This document provides information on authentication, authorization and accounting parameters and configuration for HotSpot gateway system.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip hotspot user***
Standards and Technologies: ***RADIUS***
Hardware usage: *Local traffic accounting requires additional memory*

Related Documents

- ***HotSpot Gateway***
- ***PPP User AAA***
- ***Router User AAA***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***RADIUS client***
- ***Software Package Management***
- ***IP Addresses and ARP***

Description

HotSpot User Profiles

Home menu level: ***/ip hotspot user profile***

Description

HotSpot User profiles are used for common user settings. Profiles are like user groups, they are grouping users with the same limits.

Property Description

address-pool (*name* | *none*; default: **none**) - the IP poll name which the users will be given IP addresses from. This works like dhcp-pool method in earlier versions of MikroTik RouterOS, except that it does not use DHCP, but rather the embedded one-to-one NAT

- **none** - do not reassign IP addresses to the users of this profile

advertise (yes | no; default: **no**) - whether to enable forced advertisement popups for this profile

advertise-interval (*multiple choice: time*; default: **30m,10m**) - set of intervals between showing advertisement popups. After the list is done, the last value is used for all further advertisements

advertise-timeout (*time* | *immediately* | *never*; default: **1m**) - how long to wait for advertisement to be shown, before blocking network access with walled-garden

advertise-url (*multiple choice: text*; default: **http://www.mikrotik.com/,http://www.routerboard.com/**) - list of URLs to show as advertisement popups. The list is cyclic, so when the last item reached, next time the first is shown

idle-timeout (*time* | *none*; default: **none**) - idle timeout (maximal period of inactivity) for authorized clients. It is used to detect, that client is not using outer networks (e.g. Internet), i.e., there is NO TRAFFIC coming from that client and going through the router. Reaching the timeout, user will be logged out, dropped of the host list, the address used by the user will be freed, and the session time accounted will be decreased by this value

- **none** - do not timeout idle users

incoming-filter (*name*) - name of the firewall chain applied to incoming packets from the users of this profile

incoming-packet-mark (*name*) - packet mark put on all the packets from every user of this profile automatically

keepalive-timeout (*time* | *none*; default: **00:02:00**) - keepalive timeout for authorized clients. Used to detect, that the computer of the client is alive and reachable. If check will fail during this period, user will be logged out, dropped of the host list, the address used by the user will be freed, and the session time accounted will be decreased by this value

- **none** - do not timeout unreachable users

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

name (*name*) - profile reference name

on-login (*text*; default: **""**) - script name to launch after a user has logged in

on-logout (*text*; default: **""**) - script name to launch after a user has logged out

open-status-page (*always* | *http-login*; default: **always**) - whether to show status page also for users authenticated using mac login method. Useful if you want to put some information (for example, banners or popup windows) in the alogin.html page so that all users would see it

- **http-login** - open status page only in case of http login (including cookie and https login methods)
- **always** - open http status page in case of mac login as well

outgoing-filter (*name*) - name of the firewall chain applied to outgoing packets to the users of this profile

outgoing-packet-mark (*name*) - packet mark put on all the packets to every user of this profile automatically

rate-limit (*text*; default: **""**) - Rate limitation in form of rx-rate[/tx-rate] [rx-burst-rate[/tx-burst-rate] [rx-burst-threshold[/tx-burst-threshold] [rx-burst-time[/tx-burst-time]]]]. All rates should be numbers with optional 'k' (1,000s) or 'M' (1,000,000s). If tx-rate is not specified, rx-rate is as tx-rate too. Same goes for tx-burst-rate and tx-burst-threshold and tx-burst-time. If both rx-burst-threshold and tx-burst-threshold are not specified (but burst-rate is specified), rx-rate and tx-rate is used as burst thresholds. If both rx-burst-time and tx-burst-time are not specified, 1s is used as default

session-timeout (*time*; default: **0s**) - session timeout (maximal allowed session time) for client. After this time, the user will be logged out unconditionally

- **0** - no timeout

shared-users (*integer*; default: **1**) - maximal number of simultaneously logged in users with the same username

status-autorefresh (*time* | *none*; default: **none**) - HotSpot servlet status page autorefresh interval

transparent-proxy (yes | no; default: **yes**) - whether to use transparent HTTP proxy for the authorized users of this profile

Notes

When idle-timeout or keepalive is reached, session-time for that user is reduced by the actual period of inactivity in order to prevent the user from being overcharged.

Example

HotSpot Users

Home menu level: ***/ip hotspot user***

Property Description

address (*IP address*; default: **0.0.0.0**) - static IP address. If not 0.0.0.0, client will always get the same IP address. It implies, that only one simultaneous login for that user is allowed. Any existing address will be replaced with this one using the embedded one-to-one NAT

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

bytes-in (*read-only: integer*) - total amount of bytes received from user

bytes-out (*read-only: integer*) - total amount of bytes sent to user

limit-bytes-in (*integer*; default: **0**) - maximum amount of bytes user can transmit (i.e., bytes received from the user)

- **0** - no limit

limit-bytes-out (*integer*; default: **0**) - maximum amount of bytes user can receive (i.e., bytes sent to the user)

- **0** - no limit

limit-uptime (*time*; default: **0s**) - total uptime limit for user (pre-paid time)

- **0s** - no limit

mac-address (*MAC address*; default: **00:00:00:00:00:00**) - static MAC address. If not 00:00:00:00:00:00, client is allowed to login only from that MAC address

name (*name*) - user name

packets-in (*read-only: integer*) - total amount of packets received from user (i.e., packets received from the user)

packets-out (*read-only: integer*) - total amount of packets sent to user (i.e., packets sent to the user)

password (*text*) - user password

profile (*name*; default: **default**) - user profile

routes (*text*) - routes that are to be registered on the HotSpot gateway when the client is connected. The route format is: "dst-address gateway metric" (for example, "10.1.0.0/24 10.0.0.1 1"). Several routes may be specified separated with commas

server (*name* | *all*; default: **all**) - which server is this user allowed to log in to

uptime (*read-only: time*) - total time user has been logged in

Notes

In case of **mac** authentication method, clients' MAC addresses can be used as usernames (without password)

The byte limits are total limits for each user (not for each session as at **/ip hotspot active**). So, if a user has already downloaded something, then session limit will show the total limit - (minus) already downloaded. For example, if download limit for a user is 100MB and the user has already downloaded 30MB, then session download limit after login at **/ip hotspot active** will be 100MB - 30MB = 70MB.

Should a user reach his/her limits (bytes-in >= limit-bytes-in or bytes-out >= limit-bytes-out), he/she will not be able to log in anymore.

The statistics is updated if a user is authenticated via local user database each time he/she logs out. It means, that if a user is currently logged in, then the statistics will not show current total values. Use **/ip hotspot active** submenu to view the statistics on the current user sessions.

If the user has IP address specified, only one simultaneous login is allowed. If the same credentials are used again when the user is still active, the active one will be automatically logged off.

Example

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

To add user **ex** with password **ex** that is allowed to log in only with **01:23:45:67:89:AB** MAC address and is limited to 1 hour of work:

```
[admin@MikroTik] ip hotspot user> add name=ex password=ex \
\... mac-address=01:23:45:67:89:AB limit-uptime=1h
[admin@MikroTik] ip hotspot user> print
Flags: X - disabled
 #   SERVER     NAME                      ADDRESS         PROFILE UPTIME
 0              ex                                        default 00:00:00
[admin@MikroTik] ip hotspot user> print detail
Flags: X - disabled
  0    name="ex" password="ex" mac-address=01:23:45:67:89:AB profile=default
       limit-uptime=01:00:00 uptime=00:00:00 bytes-in=0 bytes-out=0
       packets-in=0 packets-out=0
[admin@MikroTik] ip hotspot user>
```

HotSpot Active Users

Home menu level: ***/ip hotspot active***

Description

The active user list shows the list of currently logged in users. Nothing can be changed here, except user can be logged out with the **remove** command

Property Description

address (*read-only: IP address*) - IP address of the user

blocked (*read-only: flag*) - whether the user is blocked by advertisement (i.e., usual due advertisement is pending)

bytes-in (*read-only: integer*) - how many bytes did the router receive from the client

bytes-out (*read-only: integer*) - how many bytes did the router send to the client

domain (*read-only: text*) - domain of the user (if split from username)

idle-time (*read-only: time*) - the amount of time has the user been idle

idle-timeout (*read-only: time*) - the exact value of idle-timeout that applies to this user. This property shows how long should the user stay idle for it to be logged off automatically

keepalive-timeout (*read-only: time*) - the exact value of keepalive-timeout that applies to this user. This property shows how long should the user's computer stay out of reach for it to be logged off automatically

limit-bytes-in (*read-only: integer*) - maximal amount of bytes the user is allowed to send to the router

limit-bytes-out (*read-only: integer*) - maximal amount of bytes the router is allowed to send to the client

mac-address (*read-only: MAC address*) - actual MAC address of the user

packets-in (*read-only: integer*) - how many packets did the router receive from the client

packets-out (*read-only: integer*) - how many packets did the router send to the client

radius (*read-only:* yes | no) - whether the user was authenticated via RADIUS

server (*read-only: name*) - the particular server the used is logged on at.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

session-timeout (*read-only: time*) - the exact value of session-timeout that applies to this user. This property shows how long should the user stay logged-in (see uptime) for it to be logged off automatically

uptime (*read-only: time*) - current session time of the user (i.e., how long has the user been logged in)

user (*read-only: name*) - name of the user

Example

To get the list of active users:

```
[admin@MikroTik] ip hotspot active> print
Flags: R - radius, B - blocked
 #    USER            ADDRESS         UPTIME       SESSION-TIMEOUT IDLE-TIMEOUT
 0    ex              10.0.0.144      4m17s        55m43s
[admin@MikroTik] ip hotspot active>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

IP accounting

Document revision 2.1 (Fri Dec 17 18:28:01 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
Local IP Traffic Accounting
  Description
  Property Description
  Notes
  Example
  Example
Local IP Traffic Accounting Table
  Description
  Property Description
  Notes
  Example
Web Access to the Local IP Traffic Accounting Table
  Description
  Property Description
  Example

General Information

Summary

Authentication, Authorization and Accounting feature provides a possibility of local and/or remote (on RADIUS server) Point-to-Point and HotSpot user management and traffic accounting (all IP traffic passing the router is accounted).

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/user, /ppp, /ip accounting, /radius***
Standards and Technologies: ***RADIUS***
Hardware usage: *Local traffic accounting requires additional memory*

Related Documents

- ***Package Management***
- ***IP Addresses and ARP***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***HotSpot Gateway***
- ***PPP and Asynchronous Interfaces***
- ***PPPoE***
- ***PPTP***
- ***L2TP***
- ***ISDN***

Local IP Traffic Accounting

Home menu level: ***/ip accounting***

Description

As each packet passes through the router, the packet source and destination addresses are matched against an IP pair in the accounting table and the traffic for that pair is increased. The traffic of PPP, PPTP, PPPoE, ISDN and HotSpot clients can be accounted on per-user basis too. Both the number of packets and the number of bytes are accounted.

If no matching IP or user pair exists, a new entry will be added to the table

Only the packets that enter and leave the router are accounted. Packets that are dropped in the router are not counted as well as ones that are sent from the router itself. Packets that are NATted on the router will be accounted for with the actual IP addresses on each side. Packets that are going through bridged interfaces (i.e. inside the bridge interface) are also accounted correctly.

Property Description

enabled (yes | no; default: **no**) - whether local IP traffic accounting is enabled

threshold (*integer*; default: **256**) - maximum number of IP pairs in the accounting table (maximal value is 8192)

Notes

For bidirectional connections two entries will be created.

Each IP pair uses approximately 100 bytes

When the threshold limit is reached, no new IP pairs will be added to the accounting table. Each packet that is not accounted in the accounting table will then be added to the **uncounted** counter!

Example

Enable IP accounting:

```
[admin@MikroTik] ip accounting> set enabled=yes
[admin@MikroTik] ip accounting> print
      enabled: yes
    threshold: 256
[admin@MikroTik] ip accounting>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

See the uncounted packets:

```
[admin@MikroTik] ip accounting uncounted> print
    packets: 0
      bytes: 0
[admin@MikroTik] ip accounting uncounted>
```

Local IP Traffic Accounting Table

Home menu level: ***/ip accounting snapshot***

Description

When a snapshot is made for data collection, the accounting table is cleared and new IP pairs and traffic data are added. The more frequently traffic data is collected, the less likelihood that the IP pairs thereshold limit will be reached.

Property Description

bytes (*read-only: integer*) - total number of bytes, matched by this entry

dst-address (*read-only: IP address*) - destination IP address

dst-user (*read-only: text*) - recipient's name (if aplicable)

packets (*read-only: integer*) - total number of packets, matched by this entry

src-address (*read-only: IP address*) - source IP address

src-user (*read-only: text*) - sender's name (if aplicable)

Notes

Usernames are shown only if the users are connected to the router via a PPP tunnel or are authenticated by HotSpot.

Before the first snapshot is taken, the table is empty.

Example

To take a new snapshot:

```
[admin@MikroTik] ip accounting snapshot> take
[admin@MikroTik] ip accounting snapshot> print
 # SRC-ADDRESS     DST-ADDRESS     PACKETS    BYTES      SRC-USER    DST-USER
 0 192.168.0.2     159.148.172.197 474        19130
 1 192.168.0.2     10.0.0.4        3          120
 2 192.168.0.2     192.150.20.254  32         3142
 3 192.150.20.254  192.168.0.2     26         2857
 4 10.0.0.4        192.168.0.2     2          117
 5 159.148.147.196 192.168.0.2     2          136
 6 192.168.0.2     159.148.147.196 1          40
 7 159.148.172.197 192.168.0.2     835        1192962
[admin@MikroTik] ip accounting snapshot>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Web Access to the Local IP Traffic Accounting Table

Home menu level: ***/ip accounting web-access***

Description

The web page report make it possible to use the standard Unix/Linux tool wget to collect the traffic data and save it to a file or to use MikroTik shareware Traffic Counter to display the table. If the web report is enabled and the web page is viewed, the **snapshot** will be made when connection is initiated to the web page. The **snapshot** will be displayed on the web page. TCP protocol, used by http connections with the wget tool guarantees that none of the traffic data will be lost. The **snapshot** image will be made when the connection from wget is initiated. Web browsers or wget should connect to URL: **http://routerIP/accounting/ip.cgi**

Property Description

accessible-via-web (yes | no; default: **no**) - wheather the snapshot is available via web

address (*IP address* | *netmask*; default: **0.0.0.0**) - IP address range that is allowed to access the snapshot

Example

To enable web access from **10.0.0.1** server only:

```
[admin@MikroTik] ip accounting web-access> set accessible-via-web=yes \
\... address=10.0.0.1/32
[admin@MikroTik] ip accounting web-access> print
    accessible-via-web: yes
               address: 10.0.0.1/32
[admin@MikroTik] ip accounting web-access>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

PPP User AAA

Document revision 2.3 (Fri Jul 08 11:57:26 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
Local PPP User Profiles
  Description
  Property Description
  Notes
  Example
Local PPP User Database
  Description
  Property Description
  Example
Monitoring Active PPP Users
  Property Description
  Example
PPP User Remote AAA
  Property Description
  Notes
  Example

General Information

Summary

This documents provides summary, configuration reference and examples on PPP user management. This includes asynchronous PPP, PPTP, PPPoE and ISDN users.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ppp***

Related Documents

- ***Router User AAA***
- ***Software Package Management***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***IP Addresses and ARP***
- ***PPP and Asynchronous Interfaces***
- ***PPPoE***
- ***PPTP***
- ***L2TP***
- ***ISDN***

Description

The MikroTik RouterOS provides scalable Authentication, Athorization and Accounting (AAA) functionality.

Local authentication is performed using the User Database and the Profile Database. The actual configuration for the given user is composed using respective user record from the User Database, associated item from the Profile Database and the item in the Profile database which is set as default for a given service the user is authenticating to. Default profile settings from the Profile database have lowest priority while the user access record settings from the User Database have highest priority with the only exception being particular IP addresses take precedence over IP pools in the **local-address** and **remote-address** settings, which described later on.

Support for RADIUS authentication gives the ISP or network administrator the ability to manage PPP user access and accounting from one server throughout a large network. The MikroTik RouterOS has a RADIUS client which can authenticate for PPP, PPPoE, PPTP, L2TP and ISDN connections. The attributes received from RADIUS server override the ones set in the default profile, but if some parameters are not received they are taken from the respective default profile.

Local PPP User Profiles

Home menu level: ***/ppp profile***

Description

PPP profiles are used to define default values for user access records stored under **/ppp secret** submenu. Settings in **/ppp secret** User Database override corresponding **/ppp profile** settings except that single IP addresses always take precedence over IP pools when specified as **local-address** or **remote-address** parameters.

Property Description

change-tcp-mss (*yes* | *no* | *default*; default: **default**) - modifies connection MSS settings

- **yes** - adjust connection MSS value
- **no** - do not atjust connection MSS value
- **default** - accept this setting from the peer

dns-server (*IP address*) - IP address of the DNS server to supply to clients

idle-timeout (*time*) - specifies the amount of time after which the link will be terminated if there was no activity present. There is no timeout set by default

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **0s** - no link timeout is set

incoming-filter (*name*) - firewall chain name for incoming packets. Specified chain gets control for each packet coming from the client. The ppp chain should be manually added and rules with action=jump jump-target=ppp should be added to other relevant chains in order for this feature to work. For more information look at the Examples section

local-address (*IP address* | *name*) - IP address or IP address pool name for PPP server

name (*name*) - PPP profile name

only-one (*yes* | *no* | *default*; default: **default**) - defines whether a user is allowed to have more then one connection at a time

- **yes** - a user is not allowed to have more than one connection at a time
- **no** - the user is allowed to have more than one connection at a time
- **default** - accept this setting from the peer

outgoing-filter (*name*) - firewall chain name for outgoing packets. Specified chain gets control for each packet going to the client. The ppp chain should be manually added and rules with action=jump jump-target=ppp should be added to other relevant chains in order for this feature to work. For more information look at the Examples section

rate-limit (*text*; default: **""**) - rate limitation in form of rx-rate[/tx-rate] [rx-burst-rate[/tx-burst-rate] [rx-burst-threshold[/tx-burst-threshold] [rx-burst-time[/tx-burst-time]]]]. All rates are measured in bits per second, unless followed by optional 'k' suffix (kilobits per second) or 'M' suffix (megabits per second). If tx-rate is not specified, rx-rate serves as tx-rate too. The same applies for tx-burst-rate, tx-burst-threshold and tx-burst-time. If both rx-burst-threshold and tx-burst-threshold are not specified (but burst-rate is specified), rx-rate and tx-rate are used as burst thresholds. If both rx-burst-time and tx-burst-time are not specified, 1s is used as default

remote-address (*IP address* | *name*) - IP address or IP address pool name for PPP clients

session-timeout (*time*) - maximum time the connection can stay up. By default no time limit is set

- **0s** - no connection timeout

use-compression (*yes* | *no* | *default*; default: **default**) - specifies whether to use data compression or not

- **yes** - enable data compression
- **no** - disable data compression
- **default** - accept this setting from the peer

use-encryption (*yes* | *no* | *default*; default: **default**) - specifies whether to use data encryption or not

- **yes** - enable data encryption
- **no** - disable data encryption
- **default** - accept this setting from the peer

use-vj-compression (*yes* | *no* | *default*; default: **default**) - specifies whether to use Van Jacobson header compression algorithm

- **yes** - enable Van Jacobson header compression
- **no** - disable Van Jacobson header compression
- **default** - accept this setting from the peer

wins-server (*IP address*) - IP address of the WINS server to supply to Windows clients

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Notes

There are two default profiles that cannot be removed:

```
[admin@rb13] ppp profile> print
Flags: * - default
 0 * name="default" use-compression=no use-vj-compression=no use-encryption=no
only-one=no
     change-tcp-mss=yes
 1 * name="default-encryption" use-compression=default use-vj-compression=default
use-encryption=yes
     only-one=default change-tcp-mss=default
[admin@rb13] ppp profile>
```

Use Van Jacobson compression only if you have to because it may slow down the communications on bad or congested channels.

incoming-filter and **outgoing-filter** arguments add dynamic **jump** rules to chain **ppp**, where the **jump-target** argument will be equal to **incoming-filter** or **outgoing-filter** argument in **/ppp profile**. Therefore, chain **ppp** should be manually added before changing these arguments.

only-one parameter is ignored if RADIUS authentication is used

Example

To add the profile **ex** that assigns the router itself the **10.0.0.1** address, and the addresses from the **ex** pool to the clients, filtering traffic coming from clients through **mypppclients** chain:

```
[admin@rb13] ppp profile> add name=ex local-address=10.0.0.1 remote-address=ex
incoming-filter=mypppclients
[admin@rb13] ppp profile> print
Flags: * - default
 0 * name="default" use-compression=no use-vj-compression=no use-encryption=no
only-one=no
     change-tcp-mss=yes
 1   name="ex" local-address=10.0.0.1 remote-address=ex use-compression=default
     use-vj-compression=default use-encryption=default only-one=default
change-tcp-mss=default
     incoming-filter=mypppclients
 2 * name="default-encryption" use-compression=default use-vj-compression=default
use-encryption=yes
     only-one=default change-tcp-mss=default
[admin@rb13] ppp profile>
```

Local PPP User Database

Home menu level: ***/ppp secret***

Description

PPP User Database stores PPP user access records with PPP user profile assigned to each user.

Property Description

caller-id (*text*; default: "") - for PPTP and L2TP it is the IP address a client must connect from. For PPPoE it is the MAC address (written in CAPITAL letters) a client must connect from. For ISDN it is the caller's number (that may or may not be provided by the operator) the client may dial-in from

- "" - no restrictions on where clients may connect from

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

limit-bytes-in (*integer*; default: **0**) - maximal amount a client can upload, in bytes, for a session

limit-bytes-out (*integer*; default: **0**) - maximal amount a client can download, in bytes, for a session

local-address (*IP address* | *name*) - IP address or IP address pool name for PPP server

name (*name*) - user's name used for authentication

password (*text*; default: **""**) - user's password used for authentication

profile (*name*; default: **default**) - profile name to use together with this access record for user authentication

remote-address (*IP address* | *name*) - IP address or IP address pool name for PPP clients

routes (*text*) - routes that appear on the server when the client is connected. The route format is: dst-address gateway metric (for example, 10.1.0.0/ 24 10.0.0.1 1). Several routes may be specified separated with commas

service (*any* | *async* | *isdn* | *l2tp* | *pppoe* | *pptp*; default: **any**) - specifies the services available to a particular user

Example

To add the user **ex** with password **lkjrht** and profile **ex** available for PPTP service only, enter the following command:

```
[admin@rb13] ppp secret> add name=ex password=lkjrht service=pptp profile=ex
[admin@rb13] ppp secret> print
Flags: X - disabled
 #   NAME                SERVICE CALLER-ID         PASSWORD          PROFILE
REMOTE-ADDRESS
 0   ex                  pptp                      lkjrht            ex
0.0.0.0
[admin@rb13] ppp secret>
```

Monitoring Active PPP Users

Command name: ***/ppp active print***

Property Description

address (*read-only: IP address*) - IP address the client got from the server

bytes (*read-only: integer* | *integer*) - amount of bytes transfered through tis connection. First figure represents amount of transmitted traffic from the router's point of view, while the second one shows amount of received traffic

caller-id (*read-only: text*) - for PPTP and L2TP it is the IP address the client connected from. For PPPoE it is the MAC address the client connected from. For ISDN it is the caller's number the client dialed-in from

- "" - no restrictions on where clients may connect from

encoding (*read-only: text*) - shows encryption and encoding (separated with '/' if asymmetric) being used in this connection

limit-bytes-in (*read-only: integer*) - maximal amount of bytes the user is allowed to send to the router

limit-bytes-out (*read-only: integer*) - maximal amount of bytes the router is allowed to send to the client

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

name (*read-only: name*) - user name supplied at authentication stage

packets (*read-only: integer | integer*) - amount of packets transfered through tis connection. First figure represents amount of transmitted traffic from the router's point of view, while the second one shows amount of received traffic

service (*read-only: async | isdn | l2tp | pppoe | pptp*) - the type of service the user is using

session-id (*read-only: text*) - shows unique client identifier

uptime (*read-only: time*) - user's uptime

Example

```
[admin@rb13] > /ppp active print
Flags: R - radius
 #   NAME         SERVICE CALLER-ID         ADDRESS         UPTIME   ENCODING
 0   ex           pptp    10.0.11.12        10.0.0.254      1m16s    MPPE128...
[admin@rb13] > /ppp active print detail
Flags: R - radius
 0   name="ex" service=pptp caller-id="10.0.11.12" address=10.0.0.254
     uptime=1m22s encoding="MPPE128 stateless" session-id=0x8180002B
     limit-bytes-in=200000000 limit-bytes-out=0
[admin@rb13] > /ppp active print stats
Flags: R - radius
 #   NAME         BYTES                PACKETS
 0   ex           10510/159690614      187/210257
[admin@rb13] >
```

PPP User Remote AAA

Home menu level: ***/ppp aaa***

Property Description

accounting (yes | no; default: **yes**) - enable RADIUS accounting

interim-update (*time*; default: **0s**) - Interim-Update time interval

use-radius (yes | no; default: **no**) - enable user authentication via RADIUS

Notes

RADIUS user database is consulted only if the required username is not found in local user database.

Example

To enable RADIUS AAA:

```
[admin@MikroTik] ppp aaa> set use-radius=yes
[admin@MikroTik] ppp aaa> print
        use-radius: yes
        accounting: yes
    interim-update: 0s
[admin@MikroTik] ppp aaa>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

RADIUS client

Document revision 0.4 (Mon Aug 01 07:32:30 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
RADIUS Client Setup
  Description
  Property Description
  Notes
  Example
Suggested RADIUS Servers
  Description
Supported RADIUS Attributes
  Description
Troubleshooting
  Description

General Information

Summary

This document provides information about RouterOS built-in RADIUS client configuration, supported RADIUS attributes and recommendations on RADIUS server selection.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/radius***
Standards and Technologies: ***RADIUS***

Related Documents

- ***HotSpot User AAA***
- ***Router User AAA***
- ***PPP User AAA***
- ***Software Package Management***
- ***IP Addresses and ARP***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

RADIUS, short for Remote Authentication Dial-In User Service, is a remote server that provides authentication and accounting facilities to various network apliances. RADIUS authentication and accounting gives the ISP or network administrator ability to manage PPP user access and accounting from one server throughout a large network. The MikroTik RouterOS has a RADIUS client which can authenticate for HotSpot, PPP, PPPoE, PPTP, L2TP and ISDN connections. The attributes received from RADIUS server override the ones set in the default profile, but if some parameters are not received they are taken from the respective default profile.

The RADIUS server database is consulted only if no matching user acces record is found in router's local database.

Traffic is accounted locally with MikroTik Traffic Flow and Cisco **IP pairs** and snapshot image can be gathered using Syslog utilities. If RADIUS accounting is enabled, accounting information is also sent to the RADIUS server default for that service.

RADIUS Client Setup

Home menu level: ***/radius***

Description

This facility allows you to set RADIUS servers the router will use to authenticate users.

Property Description

accounting-backup (yes | no; default: **no**) - this entry is a backup RADIUS accounting server

accounting-port (*integer*; default: **1813**) - RADIUS server port used for accounting

address (*IP address*; default: **0.0.0.0**) - IP address of the RADIUS server

authentication-port (*integer*; default: **1812**) - RADIUS server port used for authentication

called-id (*text*; default: "") - value depends on Point-to-Point protocol:

- **ISDN** - phone number dialled (MSN)
- **PPPoE** - service name
- **PPTP** - server's IP address
- **L2TP** - server's IP address

domain (*text*; default: "") - Microsoft Windows domain of client passed to RADIUS servers that require domain validation

realm (*text*) - explicitly stated realm (user domain), so the users do not have to provide proper ISP domain name in user name

secret (*text*; default: "") - shared secret used to access the RADIUS server

service (*multiple choice: hotspot | login | ppp | telephony | wireless | dhcp*; default: "") - router services that will use this RADIUS server

- **hotspot** - HotSpot authentication service
- **login** - router's local user authentication

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **ppp** - Point-to-Point clients authentication
- **telephony** - IP telephony accounting
- **wireless** - wireless client authentication (client's MAC address is sent as User-Name)
- **dhcp** - DHCP protocol client authentication (client's MAC address is sent as User-Name)

timeout (*time*; default: **100ms**) - timeout after which the request should be resend

Notes

The order of the items in this list is significant.

Microsoft Windows clients send their usernames in form **domain\username**

When RADIUS server is authenticating user with CHAP, MS-CHAPv1, MS-CHAPv2, it is not using shared secret, secret is used only in authentication reply, and router is verifying it. So if you have wrong shared secret, RADIUS server will accept request, but router won't accept reply. You can see that with **/radius monitor** command, "bad-replies" number should increase whenever somebody tries to connect.

Example

To set a RADIUS server for **HotSpot** and **PPP** services that has **10.0.0.3** IP address and **ex** shared secret, you need to do the following:

```
[admin@MikroTik] radius> add service=hotspot,ppp address=10.0.0.3 secret=ex
[admin@MikroTik] radius> print
Flags: X - disabled
  #   SERVICE         CALLED-ID     DOMAIN        ADDRESS         SECRET
  0   ppp,hotspot                                 10.0.0.3        ex
[admin@MikroTik] radius>
AAA for the respective services should be enabled too:
[admin@MikroTik] radius> /ppp aaa set use-radius=yes
[admin@MikroTik] radius> /ip hotspot profile set default use-radius=yes
To view some statistics for a client:
[admin@MikroTik] radius> monitor 0
             pending: 0
            requests: 10
             accepts: 4
             rejects: 1
             resends: 15
            timeouts: 5
        bad-replies: 0
    last-request-rtt: 0s
[admin@MikroTik] radius>
```

Suggested RADIUS Servers

Description

MikroTik RouterOS RADIUS Client should work well with all RFC compliant servers. It has been tested with:

- ***FreeRADIUS***
- ***XTRadius*** (does not currently support MS-CHAP)
- ***Steel-Belted Radius***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Supported RADIUS Attributes

Description

MikroTik RADIUS Dictionaries

Here you can download ***MikroTik reference dictionary***, which incorporates all the needed RADIUS attributes. This dictionary is the minimal dictionary, which is enough to support all features of MikroTik RouterOS. It is designed for FreeRADIUS, but may also be used with many other UNIX RADIUS servers (eg. XTRadius).

Note that it may conflict with the default configuration files of RADIUS server, which have references to the Attributes, absent in this dictionary. Please correct the configuration files, not the dictionary, as no other Attributes are supported by MikroTik RouterOS.

There is also ***dictionary.mikrotik*** that can be included in an existing dictionary to support MikroTik vendor-specific Attributes.

Definitions

- **PPPs** - PPP, PPTP, PPPoE and ISDN
- **default configuration** - settings in default profile (for PPPs) or HotSpot server settings (for HotSpot)

Access-Request

- **Service-Type** - always is "Framed" (only for PPPs)
- **Framed-Protocol** - always is "PPP" (only for PPPs)
- **NAS-Identifier** - router identity
- **NAS-IP-Address** - IP address of the router itself
- **NAS-Port** - unique session ID
- **NAS-Port-Type** - async PPP - "Async"; PPTP and L2TP - "Virtual"; PPPoE - "Ethernet"; ISDN - "ISDN Sync"; HotSpot - "Ethernet | Cable | Wireless-802.11" (according to the value of nas-port-type parameter in /ip hotspot profile
- **Calling-Station-Id** - PPPoE - client MAC address with capital letters; PPTP and L2TP - client public IP address; HotSpot - MAC address of the client if it is known, or IP address of the client if MAC address is unknown; ISDN - client MSN
- **Called-Station-Id** - PPPoE - service name; PPTP and L2TP - server IP address; ISDN - interface MSN; HotSpot - MAC of the hotspot interface (if known), else IP of hotspot interface specified in hotspot menu (if set), else attribute not present
- **NAS-Port-Id** - async PPP - serial port name; PPPoE - ethernet interface name on which server is running; HotSpot - name of the hotspot interface (if known), otherwise - not present; not present for ISDN, PPTP and L2TP
- **Framed-IP-Address** - IP address of HotSpot client after Universal Client translation
- **Host-IP** - IP address of HotSpot client before Universal Client translation (the original IP

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

address of the client)

- **User-Name** - client login name
- **MS-CHAP-Domain** - User domain, if present
- **Realm** - If it is set in /radius menu, it is included in every RADIUS request as Mikrotik-Realm attribute. If it is not set, the same value is sent as in MS-CHAP-Domain attribute (if MS-CHAP-Domain is missing, Realm is not included neither)
- **User-Password** - encrypted password (used with PAP authentication)
- **CHAP-Password, CHAP-Challenge** - encrypted password and challenge (used with CHAP authentication)
- **MS-CHAP-Response, MS-CHAP-Challenge** - encrypted password and challenge (used with MS-CHAPv1 authentication)
- **MS-CHAP2-Response, MS-CHAP-Challenge** - encrypted password and challenge (used with MS-CHAPv2 authentication)

Depending on authentication methods (NOTE: HotSpot uses CHAP by default and may use also PAP if unencrypted passwords are enabled, it can not use MSCHAP):

Access-Accept

- **Framed-IP-Address** - IP address given to client. PPPs - if address belongs to 127.0.0.0/8 or 224.0.0.0/3 networks, IP pool is used from the default profile to allocate client IP address. HotSpot - used only for dhcp-pool login method (ignored for enabled-address method), if address is 255.255.255.254, IP pool is used from HotSpot settings; if Framed-IP-Address is specified, Framed-Pool is ignored
- **Framed-IP-Netmask** - client netmask. PPPs - if specified, a route will be created to the network Framed-IP-Address belongs to via the Framed-IP-Address gateway; HotSpot - ignored by HotSpot
- **Framed-Pool** - IP pool name (on the router) from which to get IP address for the client. If specified, overrides Framed-IP-Address

NOTE: if Framed-IP-Address or Framed-Pool is specified it overrides remote-address in default configuration

- **Idle-Timeout** - overrides idle-timeout in the default configuration
- **Session-Timeout** - overrides session-timeout in the default configuration
- **Max-Session-Time** - maximum session length (uptime) the user is allowed to
- **Class** - cookie, will be included in Accounting-Request unchanged
- **Framed-Route** - routes to add on the server. Format is specified in RFC2865 (Ch. 5.22), can be specified as many times as needed
- **Filter-Id** - firewall filter chain name. It is used to make a dynamic firewall rule. Firewall chain name can have suffix .in or .out, that will install rule only for incoming or outgoing traffic. Multiple Filter-id can be provided, but only last ones for incoming and outgoing is used. For PPPs - filter rules in ppp chain that will jump to the specified chain, if a packet has come to/from the client (that means that you should first create a ppp chain and make jump rules that would put actual traffic to this chain). The same applies for HotSpot, but the rules will be created in hotspot chain

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **Mark-Id** - firewall mangle chain name. The MikroTik RADIUS client upon receiving this attribute creates a dynamic firewall mangle rule with action=jump chain=hotspot and jump-target equal to the atribute value. Mangle chain name can have suffixes .in or .out, that will install rule only for incoming or outgoing traffic. Multiple Mark-id attributes can be provided, but only last ones for incoming and outgoing is used.
- **Acct-Interim-Interval** - interim-update for RADIUS client, if 0 uses the one specified in RADIUS client
- **MS-MPPE-Encryption-Policy** - require-encryption property (PPPs only)
- **MS-MPPE-Encryption-Types** - use-encryption property, non-zero value means to use encryption (PPPs only)
- **Ascend-Data-Rate** - tx/rx data rate limitation if multiple attributes are provided, first limits tx data rate, second - rx data rate. If used together with Ascend-Xmit-Rate, specifies rx rate. 0 if unlimited
- **Ascend-Xmit-Rate** - tx data rate limitation. It may be used to specify tx limit only instead of sending two sequental Ascend-Data-Rate attributes (in that case Ascend-Data-Rate will specify the receive rate). 0 if unlimited
- **MS-CHAP2-Success** - auth. response if MS-CHAPv2 was used (for PPPs only)
- **MS-MPPE-Send-Key, MS-MPPE-Recv-Key** - encryption keys for encrypted PPPs provided by RADIUS server only is MS-CHAPv2 was used as authentication (for PPPs only)
- **Ascend-Client-Gateway** - client gateway for DHCP-pool HotSpot login method (HotSpot only)
- **Recv-Limit** - total receive limit in bytes for the client
- **Xmit-Limit** - total transmit limit in bytes for the client
- **Wireless-Forward** - not forward the client's frames back to the wireless infrastructure if this attribute is set to "0" (Wireless only)
- **Wireless-Skip-Dot1x** - disable 802.1x authentication for the particulat wireless client if set to non-zero value (Wireless only)
- **Wireless-Enc-Algo** - WEP encryption algorithm: 0 - no encryption, 1 - 40-bit WEP, 2 - 104-bit WEP (Wireless only)
- **Wireless-Enc-Key** - WEP encruption key for the client (Wireless only)
- **Rate-Limit** - Datarate limitation for clients. Format is: rx-rate[/tx-rate] [rx-burst-rate[/tx-burst-rate] [rx-burst-threshold[/tx-burst-threshold] [rx-burst-time[/tx-burst-time]]]]. All rates should be numbers with optional 'k' (1,000s) or 'M' (1,000,000s). If tx-rate is not specified, rx-rate is as tx-rate too. Same goes for tx-burst-rate and tx-burst-threshold and tx-burst-time. If both rx-burst-threshold and tx-burst-threshold are not specified (but burst-rate is specified), rx-rate and tx-rate is used as burst thresholds. If both rx-burst-time and tx-burst-time are not specified, 1s is used as default.
- **Group** - Router local user group name (defines in /user group) for local users. HotSpot default profile for HotSpot users.
- **Advertise-URL** - URL of the page with advertisements that should be displayed to clients. If this attribute is specified, advertisements are enabled automatically, including transparent proxy, even if they were explicitly disabled in the corresponding user profile. Multiple attribute instances may be send by RADIUS server to specify additional URLs which are choosen in round robin fashion.
- **Advertise-Interval** - Time interval between two adjacent advertisements. Multiple attribute

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

instances may be send by RADIUS server to specify additional intervals. All interval values are threated as a list and are taken one-by-one for each successful advertisement. If end of list is reached, the last value is continued to be used.

Note that the received attributes override the default ones (set in the default profile), but if an attribute is not received from RADIUS server, the default one is to be used.

Here are some Rate-Limit examples:

- **128k** - rx-rate=128000, tx-rate=128000 (no bursts)
- **64k/128M** - rx-rate=64000, tx-rate=128000000
- **64k 256k** - rx/tx-rate=64000, rx/tx-burst-rate=256000, rx/tx-burst-threshold=64000, rx/tx-burst-time=1s
- **64k/64k 256k/256k 128k/128k 10/10** - rx/tx-rate=64000, rx/tx-burst-rate=256000, rx/tx-burst-threshold=128000, rx/tx-burst-time=10s

Accounting-Request

- **Acct-Status-Type** - Start, Stop, or Interim-Update
- **Acct-Session-Id** - accounting session ID
- **Service-Type** - same as in request (PPPs only)
- **Framed-Protocol** - same as in request (PPPs only)
- **NAS-Identifier** - same as in request
- **NAS-IP-Address** - same as in request
- **User-Name** - same as in request
- **MS-CHAP-Domain** - same as in request (only for PPPs)
- **NAS-Port-Type** - same as in request
- **NAS-Port** - same as in request
- **NAS-Port-Id** - same as in request
- **Calling-Station-Id** - same as in request
- **Called-Station-Id** - same as in request
- **Acct-Authentic** - either authenticated by the RADIUS or Local authority (PPPs only)
- **Framed-IP-Address** - IP address given to the user
- **Framed-IP-Netmask** - same as in RADIUS reply
- **Class** - RADIUS server cookie (PPPs only)
- **Acct-Delay-Time** - how long does the router try to send this Accounting-Request packet

Stop and Interim-Update Accounting-Request

- **Acct-Session-Time** - connection uptime in seconds
- **Acct-Input-Octets** - bytes received from the client
- **Acct-Input-Gigawords** - 4G (2^32) bytes received from the client (bits 32..63, when bits 0..31 are delivered in Acct-Input-Octets) (HotSpot only)
- **Acct-Input-Packets** - nubmer of packets received from the client

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **Acct-Output-Octets** - bytes sent to the client
- **Acct-Output-Gigawords** - 4G (2^32) bytes sent to the client (bits 32..63, when bits 0..31 are delivered in Acct-Output-Octets) (HotSpot only)
- **Acct-Output-Packets** - number of packets sent to the client

Stop Accounting-Request

These packets can additionally have:

- **Acct-Terminate-Cause** - session termination cause (see RFC2866 ch. 5.10)

Attribute Numeric Values

Name	VendorID	Value	RFC where it is defined
Acct-Authentic		45	RFC2866
Acct-Delay-Time		41	RFC2866
Acct-Input-Gigawords		52	RFC2869
Acct-Input-Octets		42	RFC2866
Acct-Input-Packets		47	RFC2866
Acct-Interim-Interval		85	RFC2869
Acct-Output-Gigawords		53	RFC2869
Acct-Output-Octets		43	RFC2866
Acct-Output-Packets		48	RFC2866
Acct-Session-Id		44	RFC2866
Acct-Session-Time		46	RFC2866
Acct-Status-Type		40	RFC2866
Acct-Terminate-Cause		49	RFC2866
Ascend-Client-Gateway	529	132	
Ascend-Data-Rate	529	197	
Ascend-Xmit-Rate	529	255	
Called-Station-Id		30	RFC2865
Calling-Station-Id		31	RFC2865
CHAP-Challenge		60	RFC2866
CHAP-Password		3	RFC2865
Class		25	RFC2865
Filter-Id		11	RFC2865
Framed-IP-Address		8	RFC2865

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Framed-IP-Netmask		9	RFC2865
Framed-Pool		88	RFC2869
Framed-Protocol		7	RFC2865
Framed-Route		22	RFC2865
Group	14988	3	
Idle-Timeout		28	RFC2865
MS-CHAP-Challenge	311	11	RFC2548
MS-CHAP-Domain	311	10	RFC2548
MS-CHAP-Response	311	1	RFC2548
MS-CHAP2-Response	311	25	RFC2548
MS-CHAP2-Success	311	26	RFC2548
MS-MPPE-Encryption-Policy	311	7	RFC2548
MS-MPPE-Encryption-Types	311	8	RFC2548
MS-MPPE-Recv-Key	311	17	RFC2548
MS-MPPE-Send-Key	311	16	RFC2548
NAS-Identifier		32	RFC2865
NAS-Port		5	RFC2865
NAS-Port-Id		87	RFC2869
NAS-Port-Type		61	RFC2865
Rate-Limit	14988	8	
Realm	14988	9	
Recv-Limit	14988	1	
Service-Type		6	RFC2865
Session-Timeout		27	RFC2865
User-Name		1	RFC2865
User-Password		2	RFC2865
Wireless-Enc-Algo	14988	6	
Wireless-Enc-Key	14988	7	
Wireless-Forward	14988	4	
Wireless-Skip-Dot1x	14988	5	
Xmit-Limit	14988	2	

Troubleshooting

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

- **My radius server accepts authentication request from the client with "Auth: Login OK:...", but the user cannot log on. The bad replies counter is incrementing under radius monitor**
  This situation can occur, if the radius client and server have high delay link between them. Try to increase the radius client's timeout to 600ms or more instead of the default 300ms! Also, double check, if the secrets match on client and server!

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Router User AAA

Document revision 2.3 (Fri Jul 08 11:58:32 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
Router User Groups
  Description
  Property Description
  Notes
  Example
Router Users
  Description
  Property Description
  Notes
  Example
Monitoring Active Router Users
  Description
  Property Description
  Example
Router User Remote AAA
  Description
  Property Description
  Notes
  Example

General Information

Summary

This documents provides summary, configuration reference and examples on router user management.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/user***
Hardware usage: ***Not significant***

Related Documents

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***PPP User AAA***
- ***Software Package Management***

Description

MikroTik RouterOS router user facility manage the users connecting the router from the local console, via serial terminal, telnet, SSH or Winbox. The users are authenticated using either local database or designated RADIUS server.

Each user is assigned to a user group, which denotes the rights of this user. A group policy is a combination of individual policy items.

In case the user authentication is performed using RADIUS, the RADIUS client should be previously configured under the **/radius** submenu.

Router User Groups

Home menu level: ***/user group***

Description

The router user groups provide a convenient way to assign different permissions and access rights to different user classes.

Property Description

name (*name*) - the name of the user group

policy (*multiple choice: local | telnet | ssh | ftp | reboot | read | write | policy | test | web*; default: **!local,!telnet,!ssh,!ftp,!reboot,!read,!write,!policy,!test,!web**) - group policy item set

- **local** - policy that grants rights to log in locally via console
- **telnet** - policy that grants rights to log in remotely via telnet
- **ssh** - policy that grants rights to log in remotely via secure shell protocol
- **ftp** - policy that grants remote rights to log in remotely via FTP and to transfer files from and to the router
- **reboot** - policy that allows rebooting the router
- **read** - policy that grants read access to the router's configuration. All console commands that do not alter router's configuration are allowed
- **write** - policy that grants write access to the router's configuration, except for user management. This policy does not allow to read the configuration, so make sure to enable read policy as well
- **policy** - policy that grants user management rights. Should be used together with write policy
- **test** - policy that grants rights to run ping, traceroute, bandwidth-test and wireless scan, sniffer and snooper commands
- **web** - policy that grants rights to log in remotely via WebBox
- **winbox** - policy that grants rights to log in remotely via WinBox

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **password** - policy that grants rights to change the password

Notes

There are three system groups which cannot be deleted:

```
[admin@rb13] > /user group print
 0 name="read"
policy=local,telnet,ssh,reboot,read,test,winbox,password,web,!ftp,!write,!policy

 1 name="write"
policy=local,telnet,ssh,reboot,read,write,test,winbox,password,web,!ftp,!policy

 2 name="full"
policy=local,telnet,ssh,ftp,reboot,read,write,policy,test,winbox,password,web

 3 name="test"
policy=ssh,read,policy,!local,!telnet,!ftp,!reboot,!write,!test,!winbox,!password,!web
[admin@rb13] >
```

Exclamation sign '**!**' just before policy item name means **NOT**.

Example

To add **reboot** group that is allowed to reboot the router locally or using telnet, as well as read the router's configuration, enter the following command:

```
[admin@rb13] user group> add name=reboot policy=telnet,reboot,read,local
[admin@rb13] user group> print
 0 name="read"
policy=local,telnet,ssh,reboot,read,test,winbox,password,web,!ftp,!write,!policy

 1 name="write"
policy=local,telnet,ssh,reboot,read,write,test,winbox,password,web,!ftp,!policy

 2 name="full"
policy=local,telnet,ssh,ftp,reboot,read,write,policy,test,winbox,password,web

 3 name="reboot"
policy=local,telnet,reboot,read,!ssh,!ftp,!write,!policy,!test,!winbox,!password,!web
[admin@rb13] user group>
```

Router Users

Home menu level: ***/user***

Description

Router user database stores the information such as username, password, allowed access addresses and group about router management personnel.

Property Description

address (*IP address* | *netmask*; default: **0.0.0.0/0**) - host or network address from which the user is allowed to log in

group (*name*) - name of the group the user belongs to

name (*name*) - user name. Although it must start with an alphanumeric character, it may contain "*", "_", "." and "@" symbols

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

password (*text*; default: "") - user password. If not specified, it is left blank (hit [Enter] when logging in). It conforms to standard Unix characteristics of passwords and may contain letters, digits, "*" and "_" symbols

Notes

There is one predefined user with full access rights:

```
[admin@MikroTik] user> print
Flags: X - disabled
  #   NAME                                              GROUP ADDRESS
  0   ;;; system default user
      admin                                             full  0.0.0.0/0

[admin@MikroTik] user>
```

There always should be at least one user with fulls access rights. If the user with full access rights is the only one, it cannot be removed.

Example

To add user **joe** with password **j1o2e3** belonging to **write** group, enter the following command:

```
[admin@MikroTik] user> add name=joe password=j1o2e3 group=write
[admin@MikroTik] user> print
Flags: X - disabled
  0   ;;; system default user
      name="admin" group=full address=0.0.0.0/0

  1   name="joe" group=write address=0.0.0.0/0


[admin@MikroTik] user>
```

Monitoring Active Router Users

Command name: ***/user active print***

Description

This command shows the currently active users along with respective statisics information.

Property Description

address (*read-only: IP address*) - host IP address from which the user is accessing the router

- **0.0.0.0** - the user is logged in locally from the console

name (*read-only: name*) - user name

via (*read-only: console | telnet | ssh | winbox*) - user's access method

- **console** - user is logged in locally
- **telnet** - user is logged in remotely via telnet
- **ssh** - user is logged in remotely via secure shell protocol
- **winbox** - user is logged in remotely via WinBox tool

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

when (*read-only: date*) - log in date and time

Example

To print currently active users, enter the following command:

```
[admin@rb13] user> active print
Flags: R - radius
 #   WHEN                NAME                                            ADDRESS
VIA
 0   feb/27/2004 00:41:41 admin                                           1.1.1.200
ssh
 1   feb/27/2004 01:22:34 admin                                           1.1.1.200
winbox
[admin@rb13] user>
```

Router User Remote AAA

Home menu level: ***/user aaa***

Description

Router user remote AAA enables router user authentication and accounting via RADIUS server.

Property Description

accounting (yes | no; default: **yes**) - specifies whether to use RADIUS accounting

default-group (*name*; default: **read**) - user group used by default for users authenticated via RADIUS server

interim-update (*time*; default: **0s**) - RADIUS Interim-Update interval

use-radius (yes | no; default: **no**) - specifies whether a user database on a RADIUS server should be consulted

Notes

The RADIUS user database is consulted only if the required username is not found in the local user database

Example

To enable RADIUS AAA, enter the following command:

```
[admin@MikroTik] user aaa> set use-radius=yes
[admin@MikroTik] user aaa> print
        use-radius: yes
        accounting: yes
    interim-update: 0s
     default-group: read
[admin@MikroTik] user aaa>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Traffic Flow

Document revision 1.0 (30-jun-2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Specifications
  Related Documents
  Description
  Additional Documents
General Configuration
  Description
  Property Description
Traffic-Flow Target
  Description
  Property Description
  Traffic-Flow Example

General Information

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip traffic-flow***
Hardware usage: ***Not significant***

Related Documents

- ***Cisco NetFlow***
- ***NTop***
- ***Integrating ntop with NetFlow***

Description

MikroTik Traffic-Flow is a system that provides statistic information about packets which pass through the router. Besides network monitoring and accounting, system administrators can identify various problems that may occur in the network. With help of Traffic-Flow, it is possible to analyze and optimize the overall network performance. As Traffic-Flow is compatible with Cisco NetFlow, it can be used with various utilities which are designed for Cisco's NetFlow.

Traffic-Flow supports the following NetFlow formats:

- **version 1** - the first version of NetFlow data format, do not use it, unless you have to
- **version 5** - in addition to version 1, version 5 has the BGP AS and flow sequence number

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

information included

- **version 9** - a new format which can be extended with new fields and record types thank's to its template-style design

Additional Documents

- ***Software Package Management***

General Configuration

Description

This section describes the basic configuration of Traffic-Flow.

Property Description

enabled (yes | no) - whether to enable traffic-flow service or not

interfaces (*name*) - names of those interfaces which will be used to gather statistics for traffic-flow. To specify more than one interface, separate them with a comma (",")

cache-entries (*1k* | *2k* | *4k* | *8k* | *16k* | *32k* | *64k* | *128k* | *256k* | *512k*; default: **1k**) - number of flows which can be in router's memory simultaneously

active-flow-timeout (*time*; default: **30m**) - maximum life-time of a flow

inactive-flow-timeout (*time*; default: **15s**) - how long to keep the flow active, if it is idle

Traffic-Flow Target

Description

With Traffic-Flow targets we specify those hosts which will gather the Traffic-Flow information from router.

Property Description

address (*IP address* | *port*) - IP address and port (UDP) of the host which receives Traffic-Flow statistic packets from the router

v9-template-refresh (*integer*; default: **20**) - number of packets after which the template is sent to the receiving host (only for NetFlow version 9)

v9-template-timeout - after how long to send the template, if it has not been sent

version (*1* | *5* | *9*) - which version format of NetFlow to use

General Information

Traffic-Flow Example

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

This example shows how to configure Traffic-Flow on a router

1. Enable Traffic-Flow on the router:

```
[admin@MikroTik] ip traffic-flow> set enabled=yes
[admin@MikroTik] ip traffic-flow> print
                 enabled: yes
              interfaces: all
           cache-entries: 1k
     active-flow-timeout: 30m
   inactive-flow-timeout: 15s
[admin@MikroTik] ip traffic-flow>
```

2. Specify IP address and port of the host, which will receive Traffic-Flow packets:

```
[admin@MikroTik] ip traffic-flow target> add address=192.168.0.2:2055 \
\... version=9
[admin@MikroTik] ip traffic-flow target> print
Flags: X - disabled
 #   ADDRESS              VERSION
 0   192.168.0.2:2055     9
[admin@MikroTik] ip traffic-flow target>
```

Now the router starts to send packets with Traffic-Flow information.

Some screenshots from NTop program, which has gathered Traffic-Flow information from our router and displays it in nice graphs and statistics. For example, where what kind of traffic has flown:

Top three hosts by upload and download each minute:

Overall network load each minute:

Traffic usage by each protocol:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Bandwidth Control

Document revision 1.3 (Wed Jul 20 01:32:02 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
  Additional Documents
Queue Types
  Description
  Property Description
Interface Default Queues
  Description
  Property Description
  Example
Simple Queues
  Description
  Property Description
Queue Trees
  Description
  Property Description
  Example of emulating a 128Kibps/64Kibps Line
  Queue Tree Example With Masquerading
  Equal bandwidth sharing among users

General Information

Summary

Bandwidth Control is a set of mechanisms that control data rate allocation, delay variability, timely delivery, and delivery reliability. The MikroTik RouterOS supports the following queuing disciplines:

- **PFIFO** - Packets First-In First-Out
- **BFIFO** - Bytes First-In First-Out
- **SFQ** - Stochastic Fairness Queuing
- **RED** - Random Early Detect
- **PCQ** - Per Connection Queue
- **HTB** - Hierarchical Token Bucket

Specifications

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Packages required: ***system***
License required: ***level1 (limited to 1 queue), level3***
Home menu level: ***/queue***
Standards and Technologies: ***None***
Hardware usage: *significant*

Related Documents

- ***Software Package Management***
- ***IP Addresses and ARP***
- ***Mangle***

Description

Quality of Service (QoS) means that the router should prioritize and shape network traffic. QoS is not so much about limiting, it is more about providing quality. Below are listed the some features of MikroTik RouterOS Bandwidth Control mechanism:

- limit data rate for certain IP adresses, subnets, protocols, ports, and other parameters
- limit peer-to-peer traffic
- prioritize some packet flows over others
- use queue bursts for faster WEB browsing
- apply queues on fixed time intervals
- share available traffic among users equally, or depending on the load of the channel

The queuing is applied on packets leaving the router through a real interface (i.e., the queues are applied on the outgoing interface, regarding the traffic flow), or any of the 3 additional virtual interfaces (global-in, global-out, global-total).

The QoS is performed by means of dropping packets. In case of TCP protocol, the dropped packets will be resent so there is no need to worry that with shaping we lose some TCP information.

The main terms used to describe the level of QoS for network applications, are:

- **queuing discipline (qdisc)** - an algorithm that holds and maintains a queue of packets. It specifies the order of the outgoing packets (it means that queuing discipline can reorder packets) and which packets to drop if there is no space for them
- **CIR (Committed Information Rate)** - the guaranteed data rate. It means that traffic rate, not exceeding this value should always be delivered
- **MIR (Maximal Information Rate)** - the maximal data rate router will provide
- **Priority** - the order of importance in what traffic will be processed. You can give priority to some traffic in order it to be handeled before some other traffic
- **Contention Ratio** - the ratio to which the defined data rate is shared among users (when data rate is allocated to a number of subscribers). It is the number of subscribers that have a single speed limitation, applied to all of them together. For example, the contention ratio of 1:4 means that the allocated data rate may be shared between no more than 4 users

Before sending data over an interface, it is processed with a queuing discipline. By default, queuing

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

disciplines are set under **/queue interface** for each physical interface (there is no default queuing discipline for virtual interfaces). Once we add a queue (in **/queue tree**) to a physical interface, the interface default queue, defined in **/queue interface**, for that particular interface gets ignored. It means - when a packet does not match any filter, it is sent through the interface with the highest priority.

Scheduler and Shaper qdiscs

We can classify queuing disciplines by their influence to packet flow:

- **schedulers** - queuing disciplines only reschedule packets regarding their algorithm and drop packets which 'do not fit in the queue'. Scheduler queuing disciplines are: PFIFO, BFIFO, SFQ, PCQ, RED
- **shapers** - queuing disciplines that also perform the limitation. Shapers are PCQ and HTB

Virtual Interfaces

There are 3 virtual interfaces in RouterOS, in addition to real interfaces:

- **global-in** - represents all the input interfaces in general (INGRESS queue). Please note that queues attached to global-in apply to traffic that is received by the router, before the packet filtering. global-in queueing is executed just after mangle and dst-nat
- **global-out** - represents all the output interfaces in general. Queues attached to it apply before the ones attached to a specific interface
- **global-total** - represents a virtual interface through which all the data, going through the router, is passing. When attaching a qdisc to global-total, the limitation is done in both directions. For example, if we set a total-max-limit to 256000, we will get upload+download=256kbps (maximum)

Introduction to HTB

HTB (Hierarchical Token Bucket) is a classful queuing discipline that is useful for applying different handling for different kinds of traffic. Generally, we can set only one queue for an interface, but in RouterOS queues are attached to the main Hierarchical Token Bucket (HTB) and thus have some properties derived from that parent queue. For example, we can set a maximum data rate for a workgroup and then distribute that amount of traffic between the members of that workgroup.

HTB qdisc in detail:

HTB terms:

- **queuing discipline (qdisc)** - an algorithm that holds and maintains a queue of packets. It specifies the order of the outgoing packets (it means that queuing discipline can reorder packets). Qdisc also decides which packets to drop if there is no space for them
- **filter** - a procedure that classifies packets. The filter is responsible for classifying packets so that they are put in the corresponding qdiscs
- **level** - position of a class in the hierarchy
- **inner class** - a class that has one or more child-classes attached to it. Inner classes do not store

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

any packets, but they do traffic shaping. The class also does not have its own priority

- **leaf class** - a class that has a parent but does not have any child-classes. Leaf classes are always located at level 0 of the hierarchy. Each leaf class has a qdisc, attached to it
- **self feed** - an object that represents the exit for the packets from all the classes active at its level of the hierarchy. It consists of 8 self slots
- **self slot** - an element of a self feed that corresponds to each particular priority. All classes, active at the same level, of one priority are attached to one self slot that they are using to send packets out through
- **active class (at a particular level)** - a class that is attached to a self slot at the given level
- **inner feed** - similar to self feed object, which consists of inner self slots, present on each inner class
- **inner feed slot** - similar to self slot. Each inner feed consists of inner slots which represent a priority

Each class has a parent and may have one or more children. Classes that do not have children, are put at level 0, where queues are maintained, and are called 'leaf classes'

Each class in the hierarchy can prioritize and shape traffic. There are 2 main parameters in RouterOS which refer to shaping and one - to prioritizing:

- **limit-at** - data rate that is guaranteed to a class (CIR)
- **max-limit** - maximal data rate that is allowed for a class to reach (MIR)
- **priority** - order in which classes are served at the same level (8 is the lowest priority, 1 is the highest)

Each HTB class can be in one of 3 states, depending on data rate that it consumes:

- **green** - a class the actual rate of which is equal or less than limit-at. At this state, the class is attached to self slot at the corresponding priority at its level, and is allowed to satisfy its limit-at limitation regardless of what limitations its parents have. For example, if we have a leaf class with limit-at=512000 and its parent has max-limit=limit-at=128000, the class will get its 512kbps!
- **yellow** - a class the actual rate of which is greater than limit-at and equal or less than max-limit. At this state, the class is attached to the inner slot of the corresponding priority of its parent's inner feed, which, in turn, may be attached to either its parent's inner slot of the same priority (in case the parent is also yellow), or to its own level self slot of the same priority (in case the parent is green). Upon the transition to this state, the class 'disconnects' from self feed of its level, and 'connects' to its parent's inner feed
- **red** - a class the actual rate of which exceeds max-limit. This class cannot borrow rate from its parent class

Priorities

When a leaf class wants to send some traffic (as they are the only classes that hold packets), HTB checks its priority. It will begin with the highest priority and the lowest level and proceed until the lowest priority at highest level is reached:

As you can see from the picture, leaf-classes which are at the green state, will always have a higher priority than those which are borrowing because their priority is at a lower level (level0). In this

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

picture, **Leaf1** will be served only after **Leaf2**, although it has a higher priority (7) than **Leaf1** (8).

In case of equal priorities and equal states, HTB serves these classes, using round robin algorithm.

HTB Examples

Here are some examples on how the HTB works.

Imagine the following scenario - we have 3 different kinds of traffic, marked in **/ip firewall mangle** (packet_mark1, packet_mark2 and packet_mark3), and now have bulit a HTB hierarchy:

Now let us describe some scenarios, using this HTB hierarchy.

1. Imagine a situation when there have packets arrived at Leaf1 and Leaf2. Because of this, Leaf1 attaches itself to this level's (Level 0) self slot with priority=8 and Leaf2 attaches to self slot with priority=7. Leaf3 has nothing to send, so it does nothing.
   This is a simple situation: there are active classes (Leaf1 and Leaf2) at Level 0, and as they both are in green state, they are processed in order of their priorities - at first, we serve Leaf2, then Leaf1.
2. Now assume that Leaf2 has to send more than 256kbps, for this reason, it attaches itself to its parent's (ClassB) inner feed, which recursively attaches itself to Level1 self slot at priority=7. Leaf1 continues to be at green state - it has to send packets, but not faster than 1Mbps. Leaf3 still has nothing to send.
   This is a very interesting situation because Leaf1 gets a higher priority than Leaf2 (when it is in the green state), although we have configured it for a lower priority (8) than Leaf2. It is because Leaf2 has disconnected itself from self feed at Level 0 and now is borrowing from its parent (ClassB) which has attached to self feed at Level 1. And because of this, the priority of Leaf2 'has traveled to Level1'. Remember that at first, we serve those classes which are at the lowest level with the highest priority, then continuing with the next level, and so on.
3. Consider that Leaf1 has reached its max-limit and changed its state to red, and Leaf2 now uses more than 1Mbps (and less than 2Mbps), so its parent ClassB has to borrow from ClassA and becomes yellow. Leaf3 still has no packets to send.
   This scenario shows that Leaf1 has reached its max-limit, and cannot even borrow from its parent (ClassA). Leaf2 has hierarchical reached Level2 and borrows from ClassB which recursively must borrow from ClassA because it has not enough rate available. As Leaf3 has no packets to send, the only one class who sends them, is Leaf2.
4. Assume that Leaf2 is borrowing from ClassB, ClassB from ClassA, but ClassA reaches its max-limit (2Mbps).
   In this situation Leaf2 is in yellow state, but it cannot borrow (as Class B cannot borrow from Class A).
5. Finally, let's see what happens, if Leaf1, Leaf2, Leaf3 and ClassB are in the yellow state, and ClassA is green.
   Leaf1 borrows from ClassA, Leaf2 and Leaf3 from ClassB, and ClassB also borrows from ClassA. Now all the priorities have 'moved' to Level2. So Leaf2 is on the highest priority and is served at first. As Leaf1 and Leaf3 are at the same priority (8) on the same level (2), they are served, using the round robin algorithm.

Bursts

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Bursts are used to allow higher data rates for a short period of time. Every second, the router calculates the average data rate of each class over the last **burst-time** seconds. If this average data rate is less than **burst-threshold**, burst is enabled and the actual data rate reaches **burst-limit** bps, otherwise the actual data rate falls to **max-limit** or **limit-at**.

Let us consider that we have a setup, where **max-limit**=256000, **burst-time**=8, **burst-threshold**=192000 and **burst-limit**=512000. When a user is starting to download a file via HTTP, we can observe such a situation:

At the beginning the average data rate over the last 8 seconds is 0bps because before applying the queue rule no traffic was passed, using this rule. Since this average data rate is less than **burst-threshold** (192kbps), burst is allowed. After the first second, the average data rate is (0+0+0+0+0+0+0+512)/8=64kbps, which is under **burst-threshold**. After the second second, average data rate is (0+0+0+0+0+0+512+512)/8=128kbps. After the third second comes the breakpoint when the average data rate becomes larger than **burst-threshold**. At this moment burst is disabled and the current data rate falls down to **max-limit** (256kbps).

HTB's in RouterOS

There are 4 HTB trees maintained by RouterOS:

- global-in
- globa-total
- global-out
- interface queue

When adding a simple queue, it creates 3 HTB classes (in global-in, global-total and global-out), but it does not add any classes in interface queue.

Queue tree is more flexible - you can add it to any of these HTB's.

When packet travels through the router, it passes 3 HTB trees - global-total, global-out and interface queue. If it is directed to the router, it passes only global-in HTB. If packets are sent from the router, they are applied in global-out and interface queue

Additional Documents

- ***http://linux-ip.net/articles/Traffic-Control-HOWTO/overview.html***
- ***http://luxik.cdi.cz/~devik/qos/htb/***
- ***http://www.docum.org/docum.org/docs/***

Queue Types

Home menu level: ***/queue type***

Description

In this submenu you can create your custom queue types. Afterwards, you will be able to use them in **/queue tree**, **/queue simple** or **/queue interface**.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

PFIFO and BFIFO

These queuing disciplines are based on the FIFO algorithm (First-In First-Out). The difference between PFIFO and BFIFO is that one is measured in packets and the other one in bytes. There is only one parameter called **pfifo-limit** (**bfifo-limit**) which defines how much data a FIFO queue can hold. Every packet that cannot be enqueued (if the queue is full), is dropped. Large queue sizes can increase latency.

Use FIFO queuing disciplines if you haven't a congested link

SFQ

Stochastic Fairness Queuing (SFQ) cannot limit traffic at all. Its main idea is to equalize traffic flows (TCP sessions or UDP streams) when your link is completely full.

The fairness of SFQ is ensured by hashing and round-robin algorithms. Hashing algorithm divides the session traffic over a limited number of subqueues. After **sfq-perturb** seconds the hashing algorithm changes and divides the session traffic to other subqueues. The round-robin algorithm dequeues **pcq-allot** bytes from each subqueue in a turn.

The whole SFQ queue can contain 128 packets and there are 1024 subqueues available for these packets.

Use SFQ for congested links to ensure that some connections do not starve

PCQ

To solve some SFQ imperfectness, Per Connection Queuing (PCQ) was created. It is the only classless queuing type that can do limitation. It is an improved version of SFQ without its stohastic nature. PCQ also creates subqueues, regarding the **pcq-classifier** parameter. Each subqueue has a data rate limit of **pcq-rate** and size of **pcq-limit** packets. The total size of a PCQ queue cannot be greater than **pcq-total-limit** packets.

The following example demonstrates the usage of PCQ with packets, classified by their source address.

If you classify the packets by **src-address** then all packets with different source IP addresses will be grouped into different subqueues. Now you can do the limitation or equalization for each subqueue with the **pcq-rate** parameter. Perhaps, the most significant part is to decide to which interface should we attach this queue. If we will attach it to the Local interface, all traffic from the Public interface will be grouped by src-address (probably it's not what we want), but if we attach it to the Public interface, all traffic from our clients will be grouped by src-address - so we can easily limit or equalize upload for clients.

To equalize rate among subqueues, classified by the **pcq-classifier**, set the **pcq-rate** to **0**!

PCQ can be used to dynamically equalize or shape traffic for multiple users, using little administration.

RED

Random Early Detection is a queuing mechanism which tries to avoid network congestion by

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

controlling the average queue size. When the average queue size reaches **red-min-threshold**, RED randomly chooses which arriving packet to drop. The probability how many packets will be dropped increases when the average queue size becomes larger. If the average queue size reaches **red-max-threshold**, the packets are dropped. However, there may be cases when the real queue size (not average) is much greater than **red-max-threshold**, then all packets which exceed **red-limit**are dropped.

Mainly, RED is used on congested links with high data rates. Works great with TCP protocol, but not so great with UDP.

Property Description

bfifo-limit (*integer*; default: **15000**) - maximum number of bytes that the BFIFO queue can hold

kind (*bfifo* | *pcq* | *pfifo* | *red* | *sfq*) - which queuing discipline to use

- **bfifo** - Bytes First-In, First-Out
- **pcq** - Per Connection Queue
- **pfifo** - Packets First-In, First-Out
- **red** - Random Early Detection
- **sfq** - Stohastic Fairness Queuing

name (*name*) - associative name of the queue type

pcq-classifier (*dst-address* | *dst-port* | *src-address* | *src-port*; default: "") - a classifier by which PCQ will group its subqueues. Can be used several classifiers at once, e.g., src-address,src-port will group all packets with different source address and source-ports into separate subqueues

pcq-limit (*integer*; default: **50**) - number of packets that can hold a single PCQ sub-queue

pcq-rate (*integer*; default: **0**) - maximal data rate allowed for each PCQ sub-queue. Value 0 means that there is no limitation set

pcq-total-limit (*integer*; default: **2000**) - number of packets that can hold the whole PCQ queue

pfifo-limit (*integer*) - maximum number of packets that the PFIFO queue can hold

red-avg-packet (*integer*; default: **1000**) - used by RED for average queue size calculations

red-burst (*integer*) - value in bytes which is used for determining how fast the average queue size will be influenced by the real queue size. Larger values will slow down the calculation by RED - longer bursts will be allowed

red-limit (*integer*) - value in bytes. If the real queue size (not average) exceeds this value then all packets above this value are dropped

red-max-threshold (*integer*) - value in bytes. It is the average queue size at which packet marking probability is the highest

red-min-threshold (*integer*) - average queue size in bytes. When average RED queue size reaches this value, packet marking becomes possible

sfq-allot (*integer*; default: **1514**) - amount of bytes that a subqueue is allowed to send before the next subqueue gets a turn (amount of bytes which can be sent from a subqueue in a single round-robin turn)

sfq-perturb (*integer*; default: **5**) - time in seconds. Specifies how often to change SFQ's hashing algorithm

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Interface Default Queues

Home menu level: ***/queue interface***

Description

In order to send packets over an interface, they have to be enqueued in a queue even if you do not want to limit traffic at all. Here you can specify the queue type which will be used for transmitting data.

Note that if other queues are applied for a particular packet, then these settings are not used!

Property Description

interface (*read-only: name*; default: **name of the interface**) - name of the interface

queue (*name*; default: **default**) - queue type which will be used for the interface

Example

Set the wireless interface to use **wireless-default** queue:

```
[admin@MikroTik] queue interface> set 0 queue=wireless-default
[admin@MikroTik] queue interface> print
 # INTERFACE QUEUE
 0 wlan1     wireless-default
[admin@MikroTik] queue interface>
```

Simple Queues

Description

The simpliest way to limit data rate for specific IP addresses and/or subnets, is to use simple queues.

You can also use simple queues to build advanced QoS applications. They have useful integrated features:

- Peer-to-peer traffic queuing
- Applying queue rules on chosen time intervals
- Prioritizing of traffic
- Using multiple flow marks from */ip firewall mangle*
- Shaping of bidirectional traffic (one limit for the total of upload + download)

Property Description

burst-limit (*integer* | *integer*) - maximum data rate which can be reached while the burst is active in form of in/out

burst-threshold (*integer* | *integer*) - used to calculate whether to allow burst. If the average data

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

rate over the last burst-time seconds is less than burst-threshold, the actual data rate may reach burst-limit

burst-time (*integer* | *integer*) - used to calculate average data rate

dst-address (*IP address* | *netmask*) - destination address to match

dst-netmask (*netmask*) - netmask for dst-address

flows (*text*) - packet flows which are marked in /ip firewall mangle. Current queue parameters apply only to packets which are marked with one of these flow marks

interface (*text*) - interface to which this queue applies

limit-at (*integer*) - guaranteed data rate to this queue

max-limit (*integer*) - data rate which can be reached if there is enough bandwidth available

name (*text*) - descriptive name of the queue

p2p (*any* | *all-p2p* | *bit-torrent* | *blubster* | *direct-connect* | *edonkey* | *fasttrack* | *gnutella* | *soulseek* | *winmx*) - which type of P2P traffic to match

- **all-p2p** - match all P2P traffic
- **any** - match any packet (i.e., do not check this property)

packet-marks (*name*; default: "") - packet mark to match from /ip firewall mangle. More packet marks are separated by a comma (",").

parent (*name*) - name of the parent queue in the hierarchy. Can be only other simple queue

priority (*integer*: 1..8) - priority of the queue. 1 is the highest, 8 - the lowest

queue (*name* | *name*; default: **default/default**) - name of the queue from /queue type in form of in/out

target-addresses (*IP address* | *netmask*) - limitation target IP addresses (source addresses). To use multiple addresses, separate they with comma

time (*time* | *time* | *sat* | *fri* | *thu* | *wed* | *tue* | *mon* | *sun*; default: "") - at which time and day of week to allow this queue

total-burst-limit (*integer*) - burst limit for global-total queue

total-burst-threshold (*integer*) - burst threshold for global-total queue

total-burst-time (*time*) - burst time for global-total queue

total-limit-at (*integer*) - limit-at for global-total queue (limits upload + download to total-limit-at bps)

total-max-limit (*integer*) - max-limit for global-total queue (limits upload + download to total-max-limit bps)

total-queue (*name*) - queuing discipline to use for global-total queue

Queue Trees

Home menu level: ***/queue tree***

Description

The queue trees should be used when you want to use sophisticated data rate allocation based on protocols, ports, groups of IP addresses, etc. At first you have to mark packet flows with a mark under **/ip firewall mangle** and then use this mark as an identifier for packet flows in queue trees.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Property Description

burst-limit (*integer*) - maximum data rate which can be reached while the burst is active

burst-threshold (*integer*) - used to calculate whether to allow burst. If the average data rate over the last burst-time seconds is less than burst-threshold, the actual data rate may reach burst-limit

burst-time (*time*) - used to calculate average data rate

flow (*text*) - packet flow which is marked in /ip firewall mangle. Current queue parameters apply only to packets which are marked with this flow mark

limit-at (*integer*) - guaranteed data rate to this queue

max-limit (*integer*) - data rate which can be reached if there is enough bandwidth available

name (*text*) - descriptive name for the queue

parent (*text*) - name of the parent queue. The top-level parents are the available interfaces (actually, main HTB). Lower level parents can be other queues

priority (*integer*: 1..8) - priority of the queue. 1 is the highest, 8 - the lowest

queue (*text*) - name of the queue type. Types are defined under /queue type. This parameter applies only to the leaf queues in the tree hierarchy

General Information

Example of emulating a 128Kibps/64Kibps Line

Assume, we want to emulate a 128Kibps download and 64Kibps upload line, connecting IP network **192.168.0.0/24**. The network is served through the Local interface of customer's router. The basic network setup is in the following diagram:

To solve this situation, we will use simple queues.

IP addresses on MikroTik router:

```
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   192.168.0.254/24     192.168.0.0     192.168.0.255   Local
 1   10.5.8.104/24      10.5.8.0        10.5.8.255      Public
[admin@MikroTik] ip address>
```

And routes:

```
[admin@MikroTik] ip route> print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf
 #     DST-ADDRESS        G GATEWAY         DISTANCE INTERFACE
 0 ADC 10.5.8.0/24                                   Public
 1 ADC 192.168.0.0/24                                Local
 2 A S 0.0.0.0/0          r 10.5.8.1                 Public
[admin@MikroTik] ip route>
```

Add a simple queue rule, which will limit the download traffic to 128Kib/s and upload to 64Kib/s for clients on the network **192.168.0.0/24**, served by the interface **Local**:

```
[admin@MikroTik] queue simple> add name=Limit-Local interface=Local \
\... target-address=192.168.0.0/24 max-limit=65536/131072
[admin@MikroTik] queue simple> print
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
Flags: X - disabled, I - invalid, D - dynamic
 0    name="Limit-Local" target-addresses=192.168.0.0/24 dst-address=0.0.0.0/0
      interface=Local parent=none priority=8 queue=default/default
      limit-at=0/0 max-limit=65536/131072 total-queue=default
[admin@MikroTik] queue simple>
```

The **max-limit** parameter cuts down the maximum available bandwidth. From the clients' point of view, the value **65536/131072** means that they will get maximum of 131072bps for download and 65536bps for upload. The **target-addresses** parameter defines the target network (or networks, separated by a comma) to which the queue rule will be applied.

Now see the traffic load:

```
[admin@MikroTik] interface> monitor-traffic Local
  received-packets-per-second: 7
       received-bits-per-second: 68kbps
        sent-packets-per-second: 13
           sent-bits-per-second: 135kbps

[admin@MikroTik] interface>
```

Probably, you want to exclude the server from being limited, if so, add a queoe for it without any limitation (**max-limit=0/0** which means no limitation) and move it to the beginning of the list:

```
[admin@MikroTik] queue simple> add name=Server target-addresses=192.168.0.1/32 \
\... interface=Local
[admin@MikroTik] queue simple> print
Flags: X - disabled, I - invalid, D - dynamic
 0    name="Limit-Local" target-addresses=192.168.0.0/24 dst-address=0.0.0.0/0
      interface=Local parent=none priority=8 queue=default/default
      limit-at=0/0 max-limit=65536/131072 total-queue=default

 1    name="Server" target-addresses=192.168.0.1/32 dst-address=0.0.0.0/0
      interface=Local parent=none priority=8 queue=default/default
      limit-at=0/0 max-limit=0/0 total-queue=default
[admin@MikroTik] queue simple> mo 1 0
[admin@MikroTik] queue simple> print
Flags: X - disabled, I - invalid, D - dynamic
 0    name="Server" target-addresses=192.168.0.1/32 dst-address=0.0.0.0/0
      interface=Local parent=none priority=8 queue=default/default
      limit-at=0/0 max-limit=0/0 total-queue=default

 1    name="Limit-Local" target-addresses=192.168.0.0/24 dst-address=0.0.0.0/0
      interface=Local parent=none priority=8 queue=default/default
      limit-at=0/0 max-limit=65536/131072 total-queue=default
[admin@MikroTik] queue simple>
```

Queue Tree Example With Masquerading

In the previous example we dedicated 128Kib/s download and 64Kib/s upload traffic for the local network. In this example we will guarantee 256Kib/s download (128Kib/s for the server, 64Kib/s for the Workstation and also 64Kib/s for the Laptop) and 128Kib/s for upload (64/32/32Kib/s, respectivelly) for local network devices. Additionally, if there is spare bandwidth, share it among users equally. For example, if we turn off the laptop, share its 64Kib/s download and 32Kib/s upload to the Server and Workstation.

When using masquerading, you have to mark the outgoing connection with **new-connection-mark** and take the **mark-connection** action. When it is done, you can mark all packets which belong to this connection with the **new-packet-mark** and use the **mark-packet** action.

1. At first, mark the Server's download and upload traffic. With the first rule we will mark the outgoing connection and with the second one, all packets, which belong to this connection:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] ip firewall mangle> add src-address=192.168.0.1/32 \
\... action=mark-connection new-connection-mark=server-con chain=prerouting
[admin@MikroTik] ip firewall mangle> add connection-mark=server-con \
\... action=mark-packet new-packet-mark=server chain=prerouting
[admin@MikroTik] ip firewall mangle> print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=prerouting src-address=192.168.0.1 action=mark-connection
     new-connection-mark=server-con

 1   chain=prerouting connection-mark=server-con action=mark-packet
     new-packet-mark=server
[admin@MikroTik] ip firewall mangle>
```

2. The same for Laptop and Workstation:

```
[admin@MikroTik] ip firewall mangle> add src-address=192.168.0.2 \
\... action=mark-connection new-connection-mark=lap_works-con chain=prerouting
[admin@MikroTik] ip firewall mangle> add src-address=192.168.0.3 \
\... action=mark-connection new-connection-mark=lap_works-con chain=prerouting
[admin@MikroTik] ip firewall mangle> add connection-mark=lap_works-con \
\... action=mark-packet new-packet-mark=lap_work chain=prerouting
[admin@MikroTik] ip firewall mangle> print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=prerouting src-address=192.168.0.1 action=mark-connection
     new-connection-mark=server-con

 1   chain=prerouting connection-mark=server-con action=mark-packet
     new-packet-mark=server

 2   chain=prerouting src-address=192.168.0.2 action=mark-connection
     new-connection-mark=lap_works-con

 3   chain=prerouting src-address=192.168.0.3 action=mark-connection
     new-connection-mark=lap_works-con

 4   chain=prerouting connection-mark=lap_works-con action=mark-packet
     new-packet-mark=lap_work
[admin@MikroTik] ip firewall mangle>
```

As you can see, we marked connections that belong for Laptop and Workstation with the same flow.

3. In **/queue tree** add rules that will limit Server's download and upload:

```
[admin@MikroTik] queue tree> add name=Server-Download parent=Local \
\... limit-at=131072 packet-mark=server max-limit=262144
[admin@MikroTik] queue tree> add name=Server-Upload parent=Public \
\... limit-at=65536 packet-mark=server max-limit=131072
[admin@MikroTik] queue tree> print
Flags: X - disabled, I - invalid
 0   name="Server-Download" parent=Local packet-mark=server limit-at=131072
     queue=default priority=8 max-limit=262144 burst-limit=0
     burst-threshold=0 burst-time=0s

 1   name="Server-Upload" parent=Public packet-mark=server limit-at=65536
     queue=default priority=8 max-limit=131072 burst-limit=0
     burst-threshold=0 burst-time=0s
[admin@MikroTik] queue tree>
```

And similar config for Laptop and Workstation:

```
[admin@MikroTik] queue tree> add name=Laptop-Wkst-Down parent=Local \
\... packet-mark=lap_work limit-at=65535 max-limit=262144
[admin@MikroTik] queue tree> add name=Laptop-Wkst-Up parent=Public \
\... packet-mark=lap_work limit-at=32768 max-limit=131072
[admin@MikroTik] queue tree> print
Flags: X - disabled, I - invalid
 0   name="Server-Download" parent=Local packet-mark=server limit-at=131072
     queue=default priority=8 max-limit=262144 burst-limit=0
     burst-threshold=0 burst-time=0s

 1   name="Server-Upload" parent=Public packet-mark=server limit-at=65536
     queue=default priority=8 max-limit=131072 burst-limit=0
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
      burst-threshold=0 burst-time=0s

  2   name="Laptop-Wkst-Down" parent=Local packet-mark=lap_work limit-at=65535
      queue=default priority=8 max-limit=262144 burst-limit=0
      burst-threshold=0 burst-time=0s

  3   name="Laptop-Wkst-Up" parent=Public packet-mark=lap_work limit-at=32768
      queue=default priority=8 max-limit=131072 burst-limit=0
      burst-threshold=0 burst-time=0s
[admin@MikroTik] queue tree>
```

Equal bandwidth sharing among users

This example shows how to equally share 10Mibps download and 2Mibps upload among active users in the network **192.168.0.0/24**. If **Host A** is downloading 2 Mibps, **Host B** gets 8 Mibps and vice versa. There might be situations when both hosts want to use maximum bandwidth (10 Mibps), then they will receive 5 Mibps each, the same goes for upload. This setup is also valid for more than 2 users.

At first, mark all traffic, coming from local network **192.168.0.0/24** with a mark **users**:

```
/ip firewall mangle add chain=forward src-address=192.168.0.0/24 \
   action=mark-connection new-connection-mark=users-con
/ip firewall mangle add connection-mark=users-con action=mark-packet \
   new-packet-mark=users chain=forward
```

Now we will add 2 new PCQ types. The first, called **pcq-download** will group all traffic by destination address. As we will attach this queue type to the **Local** interface, it will create a dynamic queue for each destination address (user) which is downloading to the network **192.168.0.0/24**. The second type, called **pcq-upload** will group the traffic by source address. We will attach this queue to the **Public** interface so it will make one dynamic queue for each user who is uploading to Internet from the local network **192.168.0.0/24**.

```
/queue type add name=pcq-download kind=pcq pcq-classifier=dst-address
/queue type add name=pcq-upload kind=pcq pcq-classifier=src-address
```

Finally, make a queue tree for download traffic:

```
/queue tree add name=Download parent=Local max-limit=10240000
/queue tree add parent=Download queue=pcq-download packet-mark=users
```

And for upload traffic:

```
/queue tree add name=Upload parent=Public max-limit=2048000
/queue tree add parent=Upload queue=pcq-upload packet-mark=users
```

Note! If your ISP cannot guarantee you a fixed amount of traffic, you can use just one queue for upload and one for download, attached directly to the interface:

```
/queue tree add parent=Local queue=pcq-download packet-mark=users
/queue tree add parent=Public queue=pcq-upload packet-mark=users
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Filter

Document revision 2.6 (Tue Mar 08 11:13:55 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Quick Setup Guide
  Specifications
  Related Documents
Firewall Filter
  Description
  Property Description
  Notes
Filter Applications
  Protect your RouterOS router
  Protecting the Customer's Network

General Information

Summary

The firewall implements packet filtering and thereby provides security functions that are used to manage data flow to, from and through the router. Along with the Network Address Translation it serve as a tool for preventing unauthorized access to directly attached networks and the router itself as well as a filter for outgoing traffic.

Quick Setup Guide

- To add a firewall rule which drops all **TCP** packets that are destined to port **135** and going through the router, use the following command:

```
/ip firewall filter add chain=forward dst-port=135 protocol=tcp action=drop
```

- To deny acces to the router via Telnet (protocol TCP, port 23), type the following command:

```
/ip firewall filter add chain=input protocol=tcp dst-port=23 action=drop
```

Specifications

Packages required: ***system***
License required: ***level1 (P2P filters limited to 1), level3***
Home menu level: ***/ip firewall filter***
Standards and Technologies: ***IP***, ***RFC2113***
Hardware usage: *Increases with filtering rules count*

Related Documents

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***Software Package Management***
- ***IP Addresses and ARP***
- ***NAT***
- ***Mangle***
- ***Packet Flow***

Firewall Filter

Home menu level: ***/ip firewall filter***

Description

Network firewalls keep outside threats away from sensitive data available inside the network. Whenever different networks are joined together, there is always a threat that someone from outside of your network will break into your LAN. Such break-ins may result in private data being stolen and distributed, valuable data being altered or destroyed, or entire hard drives being erased. Firewalls are used as a means of preventing or minimizing the security risks inherent in connecting to other networks. Properly configured firewall plays a key role in efficient and secure network infrastrure deployment.

MikroTik RouterOS has very powerful firewall implementation with features including:

- stateful packet filtering
- peer-to-peer protocols filtering
- traffic classification by:
  - source MAC address
  - IP addresses (network or list) and address types (broadcast, local, multicast, unicast)
  - port or port range
  - IP protocols
  - protocol options (ICMP type and code fields, TCP flags, IP options and MSS)
  - interface the packet arrived from or left through
  - internal flow and connection marks
  - ToS (DSCP) byte
  - packet content
  - rate at which packets arrive and sequence numbers
  - packet size
  - packet arrival time
  - and much more!

General Filtering Principles

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

The firewall operates by means of firewall rules. A rule is a definitive form expression that tells the router what to do with a particular IP packet. Each rule consists of two parts that are the matcher which matches traffic flow against given conditions and the action which defines what to do with the mathched packets. Rules are organized in chains for better management.

The filter facility has three default chains: **input**, **forward** and **output** that are responsible for traffic coming from, throurh and to the router, respectively. New user-defined chains can be added, as necessary. Since these chains have no default traffic to match, rules with **action=jump** and relevant **jump-target** should be added to one or more of the three default chains.

Filter Chains

As mentioned before, the firewall filtering rules are grouped together in chains. It allows a packet to be matched against one common criterion in one chain, and then passed over for processing against some other common criteria to another chain. For example a packet should be matched against the **IP address:port** pair. Of course, it could be achieved by adding as many rules with **IP address:port** match as required to the **forward** chain, but a better way could be to add one rule that matches traffic from a particular IP address, e.g.: `/ip firewall filter add src-address=1.1.1.2/32 jump-target="mychain"` and in case of successfull match passes control over the IP packet to some other chain, *id est* **mychain** in this example. Then rules that perform matching against separate ports can be added to **mychain** chain without specifying the IP addresses.

- **input** - used to process packets entering the router through one of the interfaces with the destination IP address which is one of the router's addresses. Packets passing through the router are not processed against the rules of the input chain
- **forward** - used to process packets passing through the router
- **output** - used to process packets originated from the router and leaving it through one of the interfaces. Packets passing through the router are not processed against the rules of the output chain

There are three predefined chains, which cannot be deleted:

When processing a chain, rules are taken from the chain in the order they are listed there from top to bottom. If a packet matches the criteria of the rule, then the specified action is performed on it, and no more rules are processed in that chain (the exception is the **passthrough** action). If a packet has not matched any rule within the chain, then it is accepted.

Property Description

action (*accept* | *add-dst-to-address-list* | *add-src-to-address-list* | *drop* | *jump* | *log* | *passthrough* | *reject* | *return* | *tarpit*; default: **accept**) - action to undertake if the packet matches the rule

- **accept** - accept the packet. No action is taken, i.e. the packet is passed through and no more rules are applied to it
- **add-dst-to-address-list** - adds destination address of an IP packet to the address list specified by address-list parameter
- **add-src-to-address-list** - adds source address of an IP packet to the address list specified by address-list parameter
- **drop** - silently drop the packet (without sending the ICMP reject message)
- **jump** - jump to the chain specified by the value of the jump-target parameter

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **log** - each match with this action will add a message to the system log
- **passthrough** - ignores this rule and goes on to the next one
- **reject** - reject the packet and send an ICMP reject message
- **return** - passes control back to the chain from where the jump took place
- **tarpit** - captures and holds incoming TCP connections (replies with SYN/ACK to the inbound TCP SYN packet)

address-list (*name*) - specifies the name of the address list to collect IP addresses from rules having action=add-dst-to-address-list or action=add-src-to-address-list actions. These address lists could be later used for packet matching

address-list-timeout (*time*; default: **00:00:00**) - time interval after which the address will be removed from the address list specified by address-list parameter. Used in conjunction with add-dst-to-address-list or add-src-to-address-list actions

- **00:00:00** - leave the address in the address list forever

chain (*forward* | *input* | *output* | *name*) - specifies the chain to put a particular rule into. As the different traffic is passed through different chains, always be careful in choosing the right chain for a new rule. If the input does not match the name of an already defined chain, a new chain will be created

comment (*text*) - a descriptive comment for the rule. A comment can be used to identify rules form scripts

connection-mark (*name*) - matches packets marked via mangle facility with particular connection mark

connection-bytes (*integer* | *integer*) - matches packets only if a given amount of bytes has been transfered through the particular connection

- **0** - means infinity, exempli gratia: connection-bytes=2000000-0 means that the rule matches if more than 2MB has been transfered through the relevant connection

connection-limit (*integer* | *netmask*) - restrict connection limit per address or address block

connection-state (*estabilished* | *invalid* | *new* | *related*) - interprets the connection tracking analysis data for a particular packet

- **estabilished** - a packet which belongs to an existing connection, exempli gratia a reply packet or a packet which belongs to already replied connection
- **invalid** - a packet which could not be identified for some reason. This includes out of memory condition and ICMP errors which do not correspond to any known connection. It is generally advised to drop these packets
- **new** - a packet which begins a new TCP connection
- **related** - a packet which is related to, but not part of an existing connection, such as ICMP errors or a packet which begins FTP data connection (the later requires enabled FTP connection tracking helper under /ip firewall service-port)

connection-type (*ftp* | *gre* | *h323* | *irc* | *mms* | *pptp* | *quake3* | *tftp*) - matches packets from related connections based on information from their connection tracking helpers. A relevant connection helper must be enabled under /ip firewall service-port

content (*text*) - the text packets should contain in order to match the rule

dst-address (*IP address* | *netmask* | *IP address* | *IP address*) - specifies the address range an IP packet is destined to. Note that console converts entered address/netmask value to a valid network

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

address, i.e.:1.1.1.1/24 is converted to 1.1.1.0/24

dst-address-list (*name*) - matches destination address of a packet against user-defined address list

dst-address-type (*unicast* | *local* | *broadcast* | *multicast*) - matches destination address type of the IP packet, one of the:

- **unicast** - IP addresses used for one point to another point transmission. There is only one sender and one receiver in this case
- **local** - matches addresses assigned to router's interfaces
- **broadcast** - the IP packet is sent from one point to all other points in the IP subnetwork
- **multicast** - this type of IP addressing is responsible for transmission from one or more points to a set of other points

dst-limit (*integer* | *time* | *integer* | *dst-address* | *dst-port* | *src-address* | *time*) - limits the packet per second (pps) rate on a per destination IP or per destination port base. As opposed to the limit match, every destination IP address / destination port has it's own limit. The options are as follows (in order of appearance):

- **Count** - maximum average packet rate, measured in packets per second (pps), unless followed by Time option
- **Time** - specifies the time interval over which the packet rate is measured
- **Burst** - number of packets to match in a burst
- **Mode** - the classifier(-s) for packet rate limiting
- **Expire** - specifies interval after which recorded IP addresses / ports will be deleted

dst-port (*integer*: 0..65535 | *integer*: 0..65535) - destination port number or range

packet-mark (*text*) - matches packets marked via mangle facility with particular packet mark

hotspot (*multiple choice: from-client* | *auth* | *local-dst* | *http*) - matches packets received from clients against various Hot-Spot. All values can be negated

- **from-client** - true, if a packet comes from HotSpot client
- **auth** - true, if a packet comes from authenticted client
- **local-dst** - true, if a packet has local destination IP address
- **hotspot** - true, if it is a TCP packet from client and either the transparent proxy on port 80 is enabled or the client has a proxy address configured and this address is equal to the address:port pair of the IP packet

icmp-options (*integer* | *integer*) - matches ICMP Type:Code fields

in-interface (*name*) - interface the packet has entered the router through

ipv4-options (*any* | *loose-source-routing* | *no-record-route* | *no-router-alert* | *no-source-routing* | *no-timestamp* | *none* | *record-route* | *router-alert* | *strict-source-routing* | *timestamp*) - match ipv4 header options

- **any** - match packet with at least one of the ipv4 options
- **loose-source-routing** - match packets with loose source routing option. This option is used to route the internet datagram based on information supplied by the source
- **no-record-route** - match packets with no record route option. This option is used to route the internet datagram based on information supplied by the source
- **no-router-alert** - match packets with no router alter option
- **no-source-routing** - match packets with no source routing option

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **no-timestamp** - match packets with no timestamp option
- **record-route** - match packets with record route option
- **router-alert** - match packets with router alter option
- **strict-source-routing** - match packets with strict source routing option
- **timestamp** - match packets with timestamp

jump-target (*forward* | *input* | *output* | *name*) - name of the target chain to jump to, if the action=jump is used

limit (*integer* | *time* | *integer*) - restricts packet match rate to a given limit. Usefull to reduce the amount of log messages

- **Count** - maximum average packet rate, measured in packets per second (pps), unless followed by Time option
- **Time** - specifies the time interval over which the packet rate is measured
- **Burst** - number of packets to match in a burst

log-prefix (*text*) - all messages written to logs will contain the prefix specified herein. Used in conjunction with action=log

nth (*integer* | *integer*: 0..15 | *integer*) - match a particular Nth packet received by the rule. One of 16 available counters can be used to count packets

- **Every** - match every Nth packet
- **Counter** - specifies which counter to use
- **Packet** - match on the given packet number. The value by obvious reasons must be between 0 and Every-1. If this option is used for a given counter, then there must be at least Every rules with this option, covering all values between 0 and Every-1 inclusively.

out-interface (*name*) - interface the packet will leave the router through

p2p (*all-p2p* | *bit-torrent* | *blubster* | *direct-connect* | *edonkey* | *fasttrack* | *gnutella* | *soulseek* | *warez* | *winmx*) - matches packets from various peer-to-peer (P2P) protocols

packet-size (*integer*: 0..65535 | *integer*: 0..65535) - matches packet of the specified size or size range in bytes

- **Min** - specifies lower boundary of the size range or a standalone value
- **Max** - specifies upper boundary of the size range

phys-in-interface (*name*) - matches the bridge port physical input device added to a bridge device. It is only useful if the packet has arrived through the bridge

phys-out-interface (*name*) - matches the bridge port physical output device added to a bridge device. It is only useful if the packet will leave the router through the bridge

protocol (*ddp* | *egp* | *encap* | *ggp* | *gre* | *hmp* | *icmp* | *idrp-cmtp* | *igmp* | *ipencap* | *ipip* | *ipsec-ah* | *ipsec-esp* | *iso-tp4* | *ospf* | *pup* | *rdp* | *rspf* | *st* | *tcp* | *udp* | *vmtp* | *xns-idp* | *xtp* | *integer*) - matches particular IP protocol specified by protocol name or number. You should specify this setting if you want to specify ports

psd (*integer* | *time* | *integer* | *integer*) - attempts to detect TCP and UDP scans. It is advised to assign lower weight to ports with high numbers to reduce the frequency of false positives, such as from passive mode FTP transfers

- **WeightThreshold** - total weight of the latest TCP/UDP packets with different destination ports coming from the same host to be treated as port scan sequence

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **DelayThreshold** - delay for the packets with different destination ports coming from the same host to be treated as possible port scan subsequence
- **LowPortWeight** - weight of the packets with privileged (<=1024) destination port
- **HighPortWeight** - weight of the packet with non-priviliged destination port

random (*integer*: 1..99) - matches packets randomly with given propability

reject-with (*icmp-admin-prohibited* | *icmp-echo-reply* | *icmp-host-prohibited* | *icmp-host-unreachable* | *icmp-net-prohibited* | *icmp-network-unreachable* | *icmp-port-unreachable* | *icmp-protocol-unreachable* | *tcp-reset* | *integer*) - alters the reply packet of reject action

routing-mark (*name*) - matches packets marked by mangle facility with particular routing mark

src-address (*IP address* | *netmask* | *IP address* | *IP address*) - specifies the address range an IP packet is originated from. Note that console converts entered address/netmask value to a valid network address, i.e.:1.1.1.1/24 is converted to 1.1.1.0/24

src-address-list (*name*) - matches source address of a packet against user-defined address list

src-address-type (*unicast* | *local* | *broadcast* | *multicast*) - matches source address type of the IP packet, one of the:

- **unicast** - IP addresses used for one point to another point transmission. There is only one sender and one receiver in this case
- **local** - matches addresses assigned to router's interfaces
- **broadcast** - the IP packet is sent from one point to all other points in the IP subnetwork
- **multicast** - this type of IP addressing is responsible for transmission from one or more points to a set of other points

src-mac-address (*MAC address*) - source MAC address

src-port (*integer*: 0..65535 | *integer*: 0..65535) - source port number or range

tcp-flags (*ack* | *cwr* | *ece* | *fin* | *psh* | *rst* | *syn* | *urg*) - tcp flags to match

- **ack** - acknowledging data
- **cwr** - congestion window reduced
- **ece** - ECN-echo flag (explicit congestion notification)
- **fin** - close connection
- **psh** - push function
- **rst** - drop connection
- **syn** - new connection
- **urg** - urgent data

tcp-mss (*integer*: 0..65535) - matches TCP MSS value of an IP packet

time (*time* | *time* | *sat* | *fri* | *thu* | *wed* | *tue* | *mon* | *sun*) - allows to create filter based on the packets' arrival time and date or, for locally generated packets, departure time and date

tos (*max-reliability* | *max-throughput* | *min-cost* | *min-delay* | *normal*) - specifies a match for the value of Type of Service (ToS) field of an IP header

- **max-reliability** - maximize reliability (ToS=4)
- **max-throughput** - maximize throughput (ToS=8)
- **min-cost** - minimize monetary cost (ToS=2)
- **min-delay** - minimize delay (ToS=16)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **normal** - normal service (ToS=0)

Notes

Because the NAT rules are applied first, it is important to hold this in mind when setting up firewall rules, since the original packets might be already modified by the NAT

Filter Applications

Protect your RouterOS router

To protect your router, you should not only change admin's password but also set up packet filtering. All packets with destination to the router are processed against the ip firewall input chain. Note, that the input chain does not affect packets which are being transferred through the router.

```
/ ip firewall filter
add chain=input connection-state=invalid action=drop \
        comment="Drop Invalid connections"
add chain=input connection-state=established action=accept \
        comment="Allow Established connections"
add chain=input protocol=udp action=accept \
        comment="Allow UDP"
add chain=input protocol=icmp action=accept \
        comment="Allow ICMP"
add chain=input src-address=192.168.0.0/24 action=accept \
        comment="Allow access to router from known network"
add chain=input action=drop comment="Drop anything else"
```

Protecting the Customer's Network

To protect the customer's network, we should check all traffic which goes through router and block unwanted. For icmp, tcp, udp traffic we will create chains, where will be droped all unwanted packets:

```
/ip firewall filter
add chain=forward protocol=tcp connection-state=invalid \
         action=drop comment="drop invalid connections"
add chain=forward connection-state=established action=accept \
         comment="allow already established connections"
add chain=forward connection-state=related action=accept \
         comment="allow related connections"
```

Block IP addreses called "bogons":

```
add chain=forward src-address=0.0.0.0/8 action=drop
add chain=forward dst-address=0.0.0.0/8 action=drop
add chain=forward src-address=127.0.0.0/8 action=drop
add chain=forward dst-address=127.0.0.0/8 action=drop
add chain=forward src-address=224.0.0.0/3 action=drop
add chain=forward dst-address=224.0.0.0/3 action=drop
```

Make jumps to new chains:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
add chain=forward protocol=tcp action=jump jump-target=tcp
add chain=forward protocol=udp action=jump jump-target=udp
add chain=forward protocol=icmp action=jump jump-target=icmp
```

Create tcp chain and deny some tcp ports in it:

```
add chain=tcp protocol=tcp dst-port=69 action=drop \
        comment="deny TFTP"
add chain=tcp protocol=tcp dst-port=111 action=drop \
        comment="deny RPC portmapper"
add chain=tcp protocol=tcp dst-port=135 action=drop \
        comment="deny RPC portmapper"
add chain=tcp protocol=tcp dst-port=137-139 action=drop \
        comment="deny NBT"
add chain=tcp protocol=tcp dst-port=445 action=drop \
        comment="deny cifs"
add chain=tcp protocol=tcp dst-port=2049 action=drop comment="deny NFS"
add chain=tcp protocol=tcp dst-port=12345-12346 action=drop comment="deny NetBus"
add chain=tcp protocol=tcp dst-port=20034 action=drop comment="deny NetBus"
add chain=tcp protocol=tcp dst-port=3133 action=drop comment="deny BackOriffice"
add chain=tcp protocol=tcp dst-port=67-68 action=drop comment="deny DHCP"
```

Deny udp ports in udp chain:

```
add chain=udp protocol=udp dst-port=69 action=drop comment="deny TFTP"
add chain=udp protocol=udp dst-port=111 action=drop comment="deny PRC portmapper"
add chain=udp protocol=udp dst-port=135 action=drop comment="deny PRC portmapper"
add chain=udp protocol=udp dst-port=137-139 action=drop comment="deny NBT"
add chain=udp protocol=udp dst-port=2049 action=drop comment="deny NFS"
add chain=udp protocol=udp dst-port=3133 action=drop comment="deny BackOriffice"
```

Allow only needed icmp codes in icmp chain:

```
add chain=icmp protocol=icmp icmp-options=0:0 action=accept \
        comment="drop invalid connections"
add chain=icmp protocol=icmp icmp-options=3:0 action=accept \
        comment="allow established connections"
add chain=icmp protocol=icmp icmp-options=3:1 action=accept \
        comment="allow already established connections"
add chain=icmp protocol=icmp icmp-options=4:0 action=accept \
        comment="allow source quench"
add chain=icmp protocol=icmp icmp-options=8:0 action=accept \
        comment="allow echo request"
add chain=icmp protocol=icmp icmp-options=11:0 action=accept \
        comment="allow time exceed"
add chain=icmp protocol=icmp icmp-options=12:0 action=accept \
        comment="allow parameter bad"
add chain=icmp action=drop comment="deny all other types"
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Address Lists

Document revision 2.7 (Mon May 02 10:18:10 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
Address Lists
  Description
  Property Description
  Example

General Information

Summary

Firewall address lists allow to create a list of IP addresses to be used for packet matching.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip firewall address-list***
Standards and Technologies: ***IP***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***NAT***
- ***Filter***
- ***Packet Flow***
- ***Packet Flow***

Address Lists

Description

Firewall address lists allow user to create lists of IP addresses grouped together. Firewall filter, mangle and NAT facilities can use address lists to match packets against them.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

The address list records could be updated dynamically via the **action=add-src-to-address-list** or **action=add-dst-to-address-list** items found in NAT mangle and filter facilities.

Property Description

list (*name*) - specify the name of the address list to add IP address to

address (*IP address* | *netmask* | *IP address* | *IP address*) - specify the IP address or range to be added to the address list. Note that console converts entered address/netmask value to a valid network address, i.e.:1.1.1.1/24 is converted to 1.1.1.0/24

Example

The following example creates an address list of people thet are connecting to port 23 (telnet) on the router and drops all further traffic from them. Additionaly, the address list will contain one static entry of **address=192.0.34.166/32** (www.example.com):

```
[admin@MikroTik] > /ip firewall address-list add list=drop_traffic
address=192.0.34.166/32
[admin@MikroTik] > /ip firewall address-list print
Flags: X - disabled, D - dynamic
 #   LIST         ADDRESS
 0   drop_traffic 192.0.34.166
[admin@MikroTik] > /ip firewall mangle add chain=prerouting protocol=tcp dst-port=23 \
\... action=add-src-to-address-list address-list=drop_traffic
[admin@MikroTik] > /ip firewall filter add action=drop chain=input
src-address-list=drop_traffic
[admin@MikroTik] > /ip firewall address-list print
Flags: X - disabled, D - dynamic
 #   LIST         ADDRESS
 0   drop_traffic 192.0.34.166
 1 D drop_traffic 1.1.1.1
 2 D drop_traffic 10.5.11.8
[admin@MikroTik] >
```

As seen in the output of the last **print** command, two new dynamic entries appeared in the address list. Hosts with these IP addresses tried to initialize a telnet session to the router.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Mangle

Document revision 2.8 (Tue Jul 12 09:18:22 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
Mangle
  Description
  Property Description
  Description
  Peer-to-Peer Traffic Marking
  Mark by MAC address
  Change MSS

General Information

Summary

The mangle facility allows to mark IP packets with special marks. These marks are used by various other router facilities to identify the packets. Additionaly, the mangle facility is used to modify some fields in the IP header, like TOS (DSCP) and TTL fields.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip firewall mangle***
Standards and Technologies: ***IP***
Hardware usage: *Increases with count of mangle rules*

Related Documents

- ***Software Package Management***
- ***IP Addresses and ARP***
- ***Routes, Equal Cost Multipath Routing, Policy Routing***
- ***NAT***
- ***Filter***
- ***Packet Flow***

Mangle

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Home menu level: ***/ip firewall mangle***

Description

Mangle is a kind of 'marker' that marks packets for future processing with special marks. Many other facilities in RouterOS make use of these marks, e.g. queue trees and NAT. They identify a packet based on its mark and process it accordingly. The mangle marks exist only within the router, they are not transmitted across the network.

Property Description

action (*accept* | *add-dst-to-address-list* | *add-src-to-address-list* | *change-mss* | *change-tos* | *change-ttl* | *jump* | *log* | *mark-connection* | *mark-packet* | *mark-routing* | *passthrough* | *return* | *strip-ipv4-options*; default: **accept**) - action to undertake if the packet matches the rule

- **accept** - accept the packet. No action, i.e., the packet is passed through and no more rules are applied to it
- **add-dst-to-address-list** - add destination address of an IP packet to the address list specified by address-list parameter
- **add-src-to-address-list** - add source address of an IP packet to the address list specified by address-list parameter
- **change-mss** - change Maximum Segment Size field value of the packet to a value specified by the new-mss parameter
- **change-tos** - change Type of Service field value of the packet to a value specified by the new-tos parameter
- **change-ttl** - change Time to Live field value of the packet to a value specified by the new-ttl parameter
- **jump** - jump to the chain specified by the value of the jump-target parameter
- **log** - each match with this action will add a message to the system log
- **mark-connection** - place a mark specified by the new-connection-mark parameter on the entire connection that matches the rule
- **mark-packet** - place a mark specified by the new-packet-mark parameter on a packet that matches the rule
- **mark-routing** - place a mark specified by the new-routing-mark parameter on a packet. This kind of marks is used for policy routing purposes only
- **passthrough** - ignore this rule go on to the next one
- **return** - pass control back to the chain from where the jump took place
- **strip-ipv4-options** - strip IPv4 option fields from the IP packet

address-list (*name*) - specify the name of the address list to collect IP addresses from rules having action=add-dst-to-address-list or action=add-src-to-address-list actions. These address lists could be later used for packet matching

address-list-timeout (*time*; default: **00:00:00**) - time interval after which the address will be removed from the address list specified by address-list parameter. Used in conjunction with add-dst-to-address-list or add-src-to-address-list actions

- **00:00:00** - leave the address in the address list forever

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

chain (*forward* | *input* | *output* | *postrouting* | *prerouting*) - specify the chain to put a particular rule into. As the different traffic is passed through different chains, always be careful in choosing the right chain for a new rule. If the input does not match the name of an already defined chain, a new chain will be created

comment (*text*) - free form textual comment for the rule. A comment can be used to refer the particular rule from scripts

connection-bytes (*integer* | *integer*) - match packets only if a given amount of bytes has been transfered through the particular connection

- **0** - means infinity, exempli gratia: connection-bytes=2000000-0 means that the rule matches if more than 2MB has been transfered through the relevant connection

connection-limit (*integer* | *netmask*) - restrict connection limit per address or address block

connection-mark (*name*) - match packets marked via mangle facility with particular connection mark

connection-type (*ftp* | *gre* | *h323* | *irc* | *mms* | *pptp* | *quake3* | *tftp*) - match packets from related connections based on information from their connection tracking helpers. A relevant connection helper must be enabled under /ip firewall service-port

content (*text*) - the text packets should contain in order to match the rule

dst-address (*IP address* | *netmask* | *IP address* | *IP address*) - specify the address range an IP packet is destined to. Note that console converts entered address/netmask value to a valid network address, i.e.:1.1.1.1/24 is converted to 1.1.1.0/24

dst-address-list (*name*) - match destination address of a packet against user-defined address list

dst-address-type (*unicast* | *local* | *broadcast* | *multicast*) - match destination address type of the IP packet, one of the:

- **unicast** - IP addresses used for one point to another point transmission. There is only one sender and one receiver in this case
- **local** - match addresses assigned to router's interfaces
- **broadcast** - the IP packet is sent from one point to all other points in the IP subnetwork
- **multicast** - this type of IP addressing is responsible for transmission from one or more points to a set of other points

dst-limit (*integer* | *time* | *integer* | *dst-address* | *dst-port* | *src-address* | *time*) - limit the packet per second (pps) rate on a per destination IP or per destination port base. As opposed to the limit match, every destination IP address / destination port has it's own limit. The options are as follows (in order of appearance):

- **Count** - maximum average packet rate, measured in packets per second (pps), unless followed by Time option
- **Time** - specifies the time interval over which the packet rate is measured
- **Burst** - number of packets to match in a burst
- **Mode** - the classifier(-s) for packet rate limiting
- **Expire** - specifies interval after which recorded IP addresses / ports will be deleted

dst-port (*integer*: 0..65535 | *integer*: 0..65535) - destination port number or range

hotspot (*multiple choice: from-client* | *auth* | *local-dst* | *http*) - match packets received from clients against various Hot-Spot. All values can be negated

- **from-client** - true, if a packet comes from HotSpot client

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **auth** - true, if a packet comes from authenticted client
- **local-dst** - true, if a packet has local destination IP address
- **hotspot** - true, if it is a TCP packet from client and either the transparent proxy on port 80 is enabled or the client has a proxy address configured and this address is equal to the address:port pair of the IP packet

icmp-options (*integer* | *integer*) - match ICMP Type:Code fields

in-interface (*name*) - interface the packet has entered the router through

ipv4-options (*any* | *loose-source-routing* | *no-record-route* | *no-router-alert* | *no-source-routing* | *no-timestamp* | *none* | *record-route* | *router-alert* | *strict-source-routing* | *timestamp*) - match ipv4 header options

- **any** - match packet with at least one of the ipv4 options
- **loose-source-routing** - match packets with loose source routing option. This option is used to route the internet datagram based on information supplied by the source
- **no-record-route** - match packets with no record route option. This option is used to route the internet datagram based on information supplied by the source
- **no-router-alert** - match packets with no router alter option
- **no-source-routing** - match packets with no source routing option
- **no-timestamp** - match packets with no timestamp option
- **record-route** - match packets with record route option
- **router-alert** - match packets with router alter option
- **strict-source-routing** - match packets with strict source routing option
- **timestamp** - match packets with timestamp

jump-target (*forward* | *input* | *output* | *postrouting* | *prerouting* | *name*) - name of the target chain to jump to, if the action=jump is used

limit (*integer* | *time* | *integer*) - restrict packet match rate to a given limit. Usefull to reduce the amount of log messages

- **Count** - maximum average packet rate, measured in packets per second (pps), unless followed by Time option
- **Time** - specify the time interval over which the packet rate is measured
- **Burst** - number of packets to match in a burst

log-prefix (*text*) - all messages written to logs will contain the prefix specified herein. Used in conjunction with action=log

new-connection-mark (*name*) - specify the new value of the connection mark to be used in conjunction with action=mark-connection

new-mss (*integer*) - specify MSS value to be used in conjunction with action=change-mss

new-packet-mark (*name*) - specify the new value of the packet mark to be used in conjunction with action=mark-packet

new-routing-mark (*name*) - specify the new value of the routing mark used in conjunction with action=mark-routing

new-tos (*max-reliability* | *max-throughput* | *min-cost* | *min-delay* | *normal* | *integer*) - specify TOS value to be used in conjunction with action=change-tos

- **max-reliability** - maximize reliability (ToS=4)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **max-throughput** - maximize throughput (ToS=8)
- **min-cost** - minimize monetary cost (ToS=2)
- **min-delay** - minimize delay (ToS=16)
- **normal** - normal service (ToS=0)

new-ttl (*decrement* | *increment* | *set* | *integer*) - specify the new TTL field value used in conjunction with action=change-ttl

- **decrement** - the value of the TTL field will be decremented for value
- **increment** - the value of the TTL field will be incremented for value
- **set:** - the value of the TTL field will be set to value

nth (*integer* | *integer*: 0..15 | *integer*) - match a particular Nth packet received by the rule. One of 16 available counters can be used to count packets

- **Every** - match every Nth packet
- **Counter** - specifies which counter to use
- **Packet** - match on the given packet number. The value by obvious reasons must be between 0 and Every-1. If this option is used for a given counter, then there must be at least Every rules with this option, covering all values between 0 and Every-1 inclusively.

out-interface (*name*) - match the interface name a packet left the router through

p2p (*all-p2p* | *bit-torrent* | *direct-connect* | *edonkey* | *fasttrack* | *gnutella* | *soulseek* | *warez* | *winmx*) - match packets belonging to connections of the above P2P protocols

packet-mark (*name*) - match the packets marked in mangle with specific packet mark

packet-size (*integer*: 0..65535 | *integer*: 0..65535) - matches packet of the specified size or size range in bytes

- **Min** - specifies lower boundary of the size range or a standalone value
- **Max** - specifies upper boundary of the size range

phys-in-interface (*name*) - matches the bridge port physical input device added to a bridge device. It is only useful if the packet has arrived through the bridge

protocol (*ddp* | *egp* | *encap* | *ggp* | *gre* | *hmp* | *icmp* | *idrp-cmtp* | *igmp* | *ipencap* | *ipip* | *ipsec-ah* | *ipsec-esp* | *iso-tp4* | *ospf* | *pup* | *rdp* | *rspf* | *st* | *tcp* | *udp* | *vmtp* | *xns-idp* | *xtp* | *integer*) - matches particular IP protocol specified by protocol name or number. You should specify this setting if you want to specify ports

psd (*integer* | *time* | *integer* | *integer*) - attempts to detect TCP and UDP scans. It is advised to assign lower weight to ports with high numbers to reduce the frequency of false positives, such as from passive mode FTP transfers

- **WeightThreshold** - total weight of the latest TCP/UDP packets with different destination ports coming from the same host to be treated as port scan sequence
- **DelayThreshold** - delay for the packets with different destination ports coming from the same host to be treated as possible port scan subsequence
- **LowPortWeight** - weight of the packets with privileged (<=1024) destination port
- **HighPortWeight** - weight of the packet with non-priviliged destination port

random (*integer*: 1..99) - matches packets randomly with given propability

routing-mark (*name*) - matches packets marked with the specified routing mark

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

src-address (*IP address* | *netmask* | *IP address* | *IP address*) - specifies the address range an IP packet is originated from. Note that console converts entered address/netmask value to a valid network address, i.e.:1.1.1.1/24 is converted to 1.1.1.0/24

src-address-list (*name*) - matches source address of a packet against user-defined address list

src-address-type (*unicast* | *local* | *broadcast* | *multicast*) - matches source address type of the IP packet, one of the:

- **unicast** - IP addresses used for one point to another point transmission. There is only one sender and one receiver in this case
- **local** - matches addresses assigned to router's interfaces
- **broadcast** - the IP packet is sent from one point to all other points in the IP subnetwork
- **multicast** - this type of IP addressing is responsible for transmission from one or more points to a set of other points

src-mac-address (*MAC address*) - source MAC address

src-port (*integer*: 0..65535 | *integer*: 0..65535) - source port number or range

tcp-flags (*multiple choice: ack* | *cwr* | *ece* | *fin* | *psh* | *rst* | *syn* | *urg*) - tcp flags to match

- **ack** - acknowledging data
- **cwr** - congestion window reduced
- **ece** - ECN-echo flag (explicit congestion notification)
- **fin** - close connection
- **psh** - push function
- **rst** - drop connection
- **syn** - new connection
- **urg** - urgent data

tcp-mss (*integer*: 0..65535) - matches TCP MSS value of an IP packet

time (*time* | *time* | *sat* | *fri* | *thu* | *wed* | *tue* | *mon* | *sun*) - allows to create filter based on the packets' arrival time and date or, for locally generated packets, departure time and date

tos (*max-reliability* | *max-throughput* | *min-cost* | *min-delay* | *normal*) - specifies a match for the value of Type of Service (ToS) field of an IP header

- **max-reliability** - maximize reliability (ToS=4)
- **max-throughput** - maximize throughput (ToS=8)
- **min-cost** - minimize monetary cost (ToS=2)
- **min-delay** - minimize delay (ToS=16)
- **normal** - normal service (ToS=0)

General Information

Description

The following section discusses some examples of using the mangle facility.

Peer-to-Peer Traffic Marking

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

To ensure the quality of service for network connection, interactive traffic types such as VoIP and HTTP should be prioritized over non-interactive, such as peer-to-peer network traffic. RouterOS QOS implementation uses mangle to mark different types of traffic first, and then place them into queues with different limits.

The following example enforces the P2P traffic will get no more than 1Mbps of the total link capacity when the link is heavily used by other traffic otherwice expanding to the full link capacity:

```
[admin@MikroTik] > /ip firewall mangle add chain=forward \
\... p2p=all-p2p action=mark-connection new-connection-mark=p2p_conn
[admin@MikroTik] > /ip firewall mangle add chain=forward \
\... connection-mark=p2p_conn action=mark-packet new-packet-mark=p2p
[admin@MikroTik] > /ip firewall mangle add chain=forward \
\... connection-mark=!p2p_conn action=mark-packet new-packet-mark=other
[admin@MikroTik] > /ip firewall mangle print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward p2p=all-p2p action=mark-connection new-connection-mark=p2p_conn

 1   chain=forward connection-mark=p2p_conn action=mark-packet new-packet-mark=p2p

 2   chain=forward packet-mark=!p2p_conn action=mark-packet new-packet-mark=other
[admin@MikroTik] >
[admin@MikroTik] > /queue tree add parent=Public packet-mark=p2p limit-at=1000000 \
\... max-limit=100000000 priority=8
[admin@MikroTik] > /queue tree add parent=Local packet-mark=p2p limit-at=1000000 \
\... max-limit=100000000 priority=8
[admin@MikroTik] > /queue tree add parent=Public packet-mark=other  limit-at=1000000 \
\... max-limit=100000000 priority=1
[admin@MikroTik] > /queue tree add parent=Local packet-mark=other  limit-at=1000000 \
\... max-limit=100000000 priority=1
```

Mark by MAC address

To mark traffic from a known MAC address which goes to the router or through it, do the following:

```
[admin@MikroTik] > / ip firewall mangle add chain=prerouting \
\... src-mac-address=00:01:29:60:36:E7 action=mark-connection
new-connection-mark=known_mac_conn
[admin@MikroTik] > / ip firewall mangle add chain=prerouting \
\... connection-mark=known_mac_conn action=mark-packet new-packet-mark=known_mac
```

Change MSS

It is a well known fact that VPN links have smaller packet size due to incapsulation overhead. A large packet with MSS that exceeds the MSS of the VPN link should be fragmented prior to sending it via that kind of connection. However, if the packet has DF flag set, it cannot be fragmented and should be discarded. On links that have broken path MTU discovery (PMTUD) it may lead to a number of problems, including problems with FTP and HTTP data transfer and e-mail services.

In case of link with broken PMTUD, a decrease of the MSS of the packets coming through the VPN link solves the problem. The following example demonstrates how to decrease the MSS value via mangle:

```
[admin@MikroTik] > /ip firewall mangle add out-interface=pppoe-out action=change-mss \
\.. new-mss=1300 chain=forward
[admin@MikroTik] > /ip firewall mangle print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward out-interface=pppoe-out action=change-mss new-mss=1300
[admin@MikroTik] >
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

NAT

Document revision 2.6 (Tue Apr 19 11:39:09 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
NAT
  Description
  Property Description
NAT Applications
  Description
  Example of Source NAT (Masquerading)
  Example of Destination NAT
  Example of 1:1 mapping

General Information

Summary

Network Address Translation (NAT) is a router facility that replaces source and (or) destination IP addresses of the IP packet as it pass through thhe router. It is most commonly used to enable multiple host on a private network to access the Internet using a single public IP address.

Specifications

Packages required: ***system***
License required: ***level1 (number of rules limited to 1), level3***
Home menu level: ***/ip firewall nat***
Standards and Technologies: ***IP***, ***RFC1631***, ***RFC2663***
Hardware usage: *Increases with the count of rules*

Related Documents

- ***Software Package Management***
- ***Filter***
- ***Mangle***
- ***Packet Flow***

NAT

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

Network Address Translation is an Internet standard that allows hosts on local area networks to use one set of IP addresses for internal communications and another set of IP addresses for external communications. A LAN that uses NAT is referred as *natted* network. For NAT to function, there should be a NAT gateway in each natted network. The NAT gateway (NAT router) performs IP address rewriting on the way a packet travel from/to LAN.

There are two types of NAT:

- source NAT or srcnat. This type of NAT is performed on packets that are originated from a natted network. A NAT router replaces the private source address of an IP packet with a new public IP address as it travels through the router. A reverse operation is applied to the reply packets travelling in the other direction.
- destination NAT or dstnat. This type of NAT is performed on packets that are destined to the natted network. It is most comonly used to make hosts on a private network to be acceesible from the Internet. A NAT router performing dstnat replaces the destination IP address of an IP packet as it travel through the router towards a private network.

NAT Drawbacks

Hosts behind a NAT-enabled router do not have true end-to-end connectivity. Therefore some Internet protocols might not work in scenarios with NAT. Services that require the initiation of TCP connection from outside the private network or stateless protocols such as UDP, can be disrupted. Moreover, some protocols are inherently incompatible with NAT, a bold example is AH protocol from the IPsec suite.

RouterOS includes a number of so-called NAT helpers, that enable NAT traversal for various protocols.

Redirect and Masquerade

Redirect and masquerade are special forms of destination NAT and source NAT, respectively. Redirect is similar to the regular destination NAT in the same way as masquerade is similar to the source NAT - masquerade is a special form of source NAT without need to specify **to-addresses** - outgoing interface address is used automatically. The same is for redirect - it is a form of destination NAT where **to-addresses** is not used - incoming interface address is used instead. Note that **to-ports** is meaningful for redirect rules - this is the port of the service on the router that will handle these requests (e.g. web proxy).

When packet is dst-natted (no matter - **action=nat** or **action=redirect**), dst address is changed. Information about translation of addresses (including original dst address) is kept in router's internal tables. Transparent web proxy working on router (when web requests get redirected to proxy port on router) can access this information from internal tables and get address of web server from them. If you are dst-natting to some different proxy server, it has no way to find web server's address from IP header (because dst address of IP packet that previously was address of web server has changed to address of proxy server). Starting from HTTP/1.1 there is special header in HTTP request which tells web server address, so proxy server can use it, instead of dst address of IP packet. If there is no such header (older HTTP version on client), proxy server can not determine web server address and therefore can not work.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

It means, that it is impossible to correctly transparently redirect HTTP traffic from router to some other transparent-proxy box. Only correct way is to add transparent proxy on the router itself, and configure it so that your "real" proxy is parent-proxy. In this situation your "real" proxy does not have to be transparent any more, as proxy on router will be transparent and will forward proxy-style requests (according to standard; these requests include all necessary information about web server) to "real" proxy.

Property Description

action (*accept* | *add-dst-to-address-list* | *add-src-to-address-list* | *dst-nat* | *jump* | *log* | *masquerade* | *netmap* | *passthrough* | *redirect* | *return* | *same* | *src-nat*; default: **accept**) - action to undertake if the packet matches the rule

- **accept** - accepts the packet. No action is taken, i.e. the packet is passed through and no more rules are applied to it
- **add-dst-to-address-list** - adds destination address of an IP packet to the address list specified by address-list parameter
- **add-src-to-address-list** - adds source address of an IP packet to the address list specified by address-list parameter
- **dst-nat** - replaces destination address of an IP packet to values specified by to-addresses and to-ports parameters
- **jump** - jump to the chain specified by the value of the jump-target parameter
- **log** - each match with this action will add a message to the system log
- **masquerade** - replaces source address of an IP packet to an automatically determined by the routing facility IP address
- **netmap** - creates a static 1:1 mapping of one set of IP addresses to another one. Often used to distribute public IP addresses to hosts on private networks
- **passthrough** - ignores this rule goes on to the next one
- **redirect** - replaces destination address of an IP packet to one of the router's local addresses
- **return** - passes control back to the chain from where the jump took place
- **same** - gives a particular client the same source/destination IP address from supplied range for each connection. This is most frequently used for services that expect the same client address for multiple connections from the same client
- **src-nat** - replaces source address of an IP packet to values specified by to-addresses and to-ports parameters

address-list (*name*) - specifies the name of the address list to collect IP addresses from rules having action=add-dst-to-address-list or action=add-src-to-address-list actions. These address lists could be later used for packet matching

address-list-timeout (*time*; default: **00:00:00**) - time interval after which the address will be removed from the address list specified by address-list parameter. Used in conjunction with add-dst-to-address-list or add-src-to-address-list actions

- **00:00:00** - leave the address in the address list forever

chain (*dstnat* | *srcnat* | *name*) - specifies the chain to put a particular rule into. As the different traffic is passed through different chains, always be careful in choosing the right chain for a new rule. If the input does not match the name of an already defined chain, a new chain will be created

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **dstnat** - a rule placed in this chain is applied after routing. The rules that replace destination addresses of IP packets should be placed there
- **srcnat** - a rule placed in this chain is applied before routing. The rules that replace the source addresses of IP packets should be placed there

comment (*text*) - a descriptive comment for the rule. A comment can be used to identify rules form scripts

connection-bytes (*integer* | *integer*) - matches packets only if a given amount of bytes has been transfered through the particular connection

- **0** - means infinity, exempli gratia: connection-bytes=2000000-0 means that the rule matches if more than 2MB has been transfered through the relevant connection

connection-limit (*integer* | *netmask*) - restrict connection limit per address or address block

connection-mark (*name*) - matches packets marked via mangle facility with particular connection mark

connection-type (*ftp* | *gre* | *h323* | *irc* | *mms* | *pptp* | *quake3* | *tftp*) - matches packets from related connections based on information from their connection tracking helpers. A relevant connection helper must be enabled under /ip firewall service-port

content (*text*) - the text packets should contain in order to match the rule

dst-address (*IP address* | *netmask* | *IP address* | *IP address*) - specifies the address range an IP packet is destined to. Note that console converts entered address/netmask value to a valid network address, i.e.:1.1.1.1/24 is converted to 1.1.1.0/24

dst-address-list (*name*) - matches destination address of a packet against user-defined address list

dst-address-type (*unicast* | *local* | *broadcast* | *multicast*) - matches destination address type of the IP packet, one of the:

- **unicast** - IP addresses used for one point to another point transmission. There is only one sender and one receiver in this case
- **local** - matches addresses assigned to router's interfaces
- **broadcast** - the IP packet is sent from one point to all other points in the IP subnetwork
- **multicast** - this type of IP addressing is responsible for transmission from one or more points to a set of other points

dst-limit (*integer* | *time* | *integer* | *dst-address* | *dst-port* | *src-address* | *time*) - limits the packet per second (pps) rate on a per destination IP or per destination port base. As opposed to the limit match, every destination IP address / destination port has it's own limit. The options are as follows (in order of appearance):

- **Count** - maximum average packet rate, measured in packets per second (pps), unless followed by Time option
- **Time** - specifies the time interval over which the packet rate is measured
- **Burst** - number of packets to match in a burst
- **Mode** - the classifier(-s) for packet rate limiting
- **Expire** - specifies interval after which recorded IP addresses / ports will be deleted

dst-port (*integer*: 0..65535 | *integer*: 0..65535) - destination port number or range

hotspot (*multiple choice: from-client* | *auth* | *local-dst*) - matches packets received from clients against various Hot-Spot. All values can be negated

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **from-client** - true, if a packet comes from HotSpot client
- **auth** - true, if a packet comes from authenticted client
- **local-dst** - true, if a packet has local destination IP address

icmp-options (*integer* | *integer*) - matches ICMP Type:Code fields

in-interface (*name*) - interface the packet has entered the router through

ipv4-options (*any* | *loose-source-routing* | *no-record-route* | *no-router-alert* | *no-source-routing* | *no-timestamp* | *none* | *record-route* | *router-alert* | *strict-source-routing* | *timestamp*) - match ipv4 header options

- **any** - match packet with at least one of the ipv4 options
- **loose-source-routing** - match packets with loose source routing option. This option is used to route the internet datagram based on information supplied by the source
- **no-record-route** - match packets with no record route option. This option is used to route the internet datagram based on information supplied by the source
- **no-router-alert** - match packets with no router alter option
- **no-source-routing** - match packets with no source routing option
- **no-timestamp** - match packets with no timestamp option
- **record-route** - match packets with record route option
- **router-alert** - match packets with router alter option
- **strict-source-routing** - match packets with strict source routing option
- **timestamp** - match packets with timestamp

jump-target (*dstnat* | *srcnat* | *name*) - name of the target chain to jump to, if the action=jump is used

limit (*integer* | *time* | *integer*) - restricts packet match rate to a given limit. Usefull to reduce the amount of log messages

- **Count** - maximum average packet rate, measured in packets per second (pps), unless followed by Time option
- **Time** - specifies the time interval over which the packet rate is measured
- **Burst** - number of packets to match in a burst

log-prefix (*text*) - all messages written to logs will contain the prefix specified herein. Used in conjunction with action=log

nth (*integer* | *integer*: 0..15 | *integer*) - match a particular Nth packet received by the rule. One of 16 available counters can be used to count packets

- **Every** - match every Nth packet
- **Counter** - specifies which counter to use
- **Packet** - match on the given packet number. The value by obvious reasons must be between 0 and Every-1. If this option is used for a given counter, then there must be at least Every rules with this option, covering all values between 0 and Every-1 inclusively.

out-interface (*name*) - interface the packet is leaving the router through

packet-mark (*text*) - matches packets marked via mangle facility with particular packet mark

packet-size (*integer*: 0..65535 | *integer*: 0..65535) - matches packet of the specified size or size range in bytes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **Min** - specifies lower boundary of the size range or a standalone value
- **Max** - specifies upper boundary of the size range

phys-in-interface (*name*) - matches the bridge port physical input device added to a bridge device. It is only useful if the packet has arrived through the bridge

phys-out-interface (*name*) - matches the bridge port physical output device added to a bridge device. It is only useful if the packet will leave the router through the bridge

protocol (*ddp* | *egp* | *encap* | *ggp* | *gre* | *hmp* | *icmp* | *idrp-cmtp* | *igmp* | *ipencap* | *ipip* | *ipsec-ah* | *ipsec-esp* | *iso-tp4* | *ospf* | *pup* | *rdp* | *rspf* | *st* | *tcp* | *udp* | *vmtp* | *xns-idp* | *xtp* | *integer*) - matches particular IP protocol specified by protocol name or number. You should specify this setting if you want to specify ports

psd (*integer* | *time* | *integer* | *integer*) - attempts to detect TCP and UDP scans. It is advised to assign lower weight to ports with high numbers to reduce the frequency of false positives, such as from passive mode FTP transfers

- **WeightThreshold** - total weight of the latest TCP/UDP packets with different destination ports coming from the same host to be treated as port scan sequence
- **DelayThreshold** - delay for the packets with different destination ports coming from the same host to be treated as possible port scan subsequence
- **LowPortWeight** - weight of the packets with privileged (<=1024) destination port
- **HighPortWeight** - weight of the packet with non-priviliged destination port

random (*integer*) - match packets randomly with given propability

routing-mark (*name*) - matches packets marked by mangle facility with particular routing mark

same-not-by-dst (*yes* | *no*) - specifies whether to account or not to account for destination IP address when selecting a new source IP address for packets matched by rules with action=same

src-address (*IP address* | *netmask* | *IP address* | *IP address*) - specifies the address range an IP packet is originated from. Note that console converts entered address/netmask value to a valid network address, i.e.:1.1.1.1/24 is converted to 1.1.1.0/24

src-address-list (*name*) - matches source address of a packet against user-defined address list

src-address-type (*unicast* | *local* | *broadcast* | *multicast*) - matches source address type of the IP packet, one of the:

- **unicast** - IP addresses used for one point to another point transmission. There is only one sender and one receiver in this case
- **local** - matches addresses assigned to router's interfaces
- **broadcast** - the IP packet is sent from one point to all other points in the IP subnetwork
- **multicast** - this type of IP addressing is responsible for transmission from one or more points to a set of other points

src-mac-address (*MAC address*) - source MAC address

src-port (*integer*: 0..65535 | *integer*: 0..65535) - source port number or range

tcp-mss (*integer*: 0..65535) - matches TCP MSS value of an IP packet

time (*time* | *time* | *sat* | *fri* | *thu* | *wed* | *tue* | *mon* | *sun*) - allows to create filter based on the packets' arrival time and date or, for locally generated packets, departure time and date

to-addresses (*IP address* | *IP address*; default: **0.0.0.0**) - address or address range to replace original address of an IP packet with

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

to-ports (*integer*: 0..65535 | *integer*: 0..65535) - port or port range to replace original port of an IP packet with

tos (*max-reliability* | *max-throughput* | *min-cost* | *min-delay* | *normal*) - specifies a match to the value of Type of Service (ToS) field of IP header

- **max-reliability** - maximize reliability (ToS=4)
- **max-throughput** - maximize throughput (ToS=8)
- **min-cost** - minimize monetary cost (ToS=2)
- **min-delay** - minimize delay (ToS=16)
- **normal** - normal service (ToS=0)

NAT Applications

Description

In this section some NAT applications and examples of them are discussed.

Basic NAT configuration

Assume we want to create router that:

- "hides" the private LAN "behind" one address
- provides Public IP to the Local server
- creates 1:1 mapping of network addresses

Example of Source NAT (Masquerading)

If you want to "hide" the private LAN 192.168.0.0/24 "behind" one address 10.5.8.109 given to you by the ISP, you should use the source network address translation (masquerading) feature of the MikroTik router. The masquerading will change the source IP address and port of the packets originated from the network 192.168.0.0/24 to the address 10.5.8.109 of the router when the packet is routed through it.

To use masquerading, a source NAT rule with action 'masquerade' should be added to the firewall configuration:

```
/ip firewall nat add chain=srcnat action=masquerade out-interface=Public
```

All outgoing connections from the network 192.168.0.0/24 will have source address 10.5.8.109 of the router and source port above 1024. No access from the Internet will be possible to the Local addresses. If you want to allow connections to the server on the local network, you should use destination Network Address Translation (NAT).

Example of Destination NAT

If you want to link Public IP 10.5.8.200 address to Local one 192.168.0.109, you should use

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

destination address translation feature of the MikroTik router. Also if you want allow Local server to talk with outside with given Public IP you should use source address translation, too

Add Public IP to Public interface:

```
/ip address add address=10.5.8.200/32 interface=Public
```

Add rule allowing access to the internal server from external networks:

```
/ip firewall nat add chain=dstnat dst-address=10.5.8.200 action=dst-nat \
        to-addresses=192.168.0.109
```

Add rule allowing the internal server to talk to the outer networks having its source address translated to 10.5.8.200:

```
/ip firewall nat add chain=srcnat src-address=192.168.0.109 action=src-nat \
        to-addresses=10.5.8.200
```

Example of 1:1 mapping

If you want to link Public IP subnet 11.11.11.0/24 to local one 2.2.2.0/24, you should use destination address translation and source address translation features with **action=netmap**.

```
/ip firewall nat add chain=dstnat dst-address=11.11.11.1-11.11.11.254 \
        action=netmap to-addresses=2.2.2.1-2.2.2.254

/ip firewall nat add chain=srcnat src-address=2.2.2.1-2.2.2.254 \
        action=netmap to-addresses=11.11.11.1-11.11.11.254
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Packet Flow

Document revision 2.6 (Tue Jun 14 17:24:04 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
Packet Flow
  Description
General Firewall Information
  Description

General Information

Summary

This manual describes the order in which an IP packet traverses various internal facilities of the router and some general information regarding packet handling, common IP protocols and protocol options.

Specifications

Packages required: ***system***
License required: ***level3***
Home menu level: ***/ip firewall***
Standards and Technologies: ***IP***
Hardware usage: *Increases with NAT, mangle and filter rules count*

Related Documents

- ***Software Package Management***
- ***NAT***
- ***Mangle***
- ***Filter***

Packet Flow

Description

MikroTik RouterOS is designed to be easy to operate in various aspects, including IP firewall.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Therefore regular firewall policies can be created and deployed without the knowledge about how the packets are processed in the router. For example, if all that required is just natting internal clients to a public address, the following command can be issued (assuming the interface to the Internet in named **Public**):

```
/ip firewall nat add action=masquerade out-interface=Public chain=srcnat
```

Regular packet filtering, bandwith management or packet marking can be configured with ease in a similar manner. However, a more complicated configuration could be deployed only with a good understanding of the underlying processes in the router.

The packet flow through the router is depicted in the following diagram:

As can be seen on the diagram, there are five chains in the processing pipeline. These are **prerouting**, **input**, **forward**, **output** and **postrouting**. The actions performed on a packet in each chain are discussed later in this chapter.

A paket can enter processing conveyer of the router in two ways. First, a packet can come from one of the interfaces present in the roter (then the interface is referred as **input interface**). Second, it can be originated from a local process, like web proxy, VPN or others. Alike, there are two ways for a packet to leave the processing pipeline. A packet can leave through the one of the router's interfaces (in this case the interface is referred as **output interface**) or it can end up in the local process. In general, traffic can be destined to one of the router's IP addresses, it can originate from the router or simply should be passed through. To further complicate things the traffic can be bridged or routed one, which is determined during the **Bridge Decision** stage.

Routed traffic

The traffic which is being routed can be one of three types:

- the traffic which is destined to the router itself. The IP packets has destination address equal to one of the router's IP addresses. A packet enters the router through the **input interface**, sequentially traverses **prerouting** and **input** chains and ends up in the local process. Consequently, a packet can be filtered in the **input** chain filter and mangled in two places: the **input** and the **prerouting** chain filters.
- the traffic originated by the router. In this case the IP packets have their source addresses identical to one of the router's IP addresses. Such packets travel through the **output** chain, then they are passed to the routing facility where an appropriate routing path for each packet is determined and leave through the **postrouting** chain.
- one which passes through the router. These packets go through the **prerouting**, **forward** and **postrouting** chains.

The actions imposed by various router facilities are sequentially applied to a packet in each of the default chains. The exact order they are applied is pictured in the bottom of the flow diagram. *Exempli gratia*, for a packet passing **postrouting** chain the mangle rules are applied first, two types of queuing come in second place and finally source NAT is performed on packets that need to be natted.

Note, that a given packet can come through only one of the **input**, **forward** or **output** chains.

Bridged Traffic

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

In case the incoming traffic needs to be bridged (do not confuse it with the traffic coming from the bridge interface, which should be routed) it is first determined whether it is an IP traffic or not. After that the IP traffic goes through the **prerouting**, **forward** and **postrouting** chains, while non-IP traffic goes directly to the interface queue. Both types of traffic, however, undergo the bridge firewall check first.

Additional arrows from IPsec boxes shows the processing of encrypted packets (they need to be encrypted / decrypted first and then processed as usual, *id est* from the point an ordinal packet enters the router).

If the packet is bridged one, the 'Routing Decision' changes to 'Bridge Forwarding Decision'. In case the bridge is forwarding non-IP packets, all things regarding IP protocol are not applicable ('Universal Client', 'Conntrack', 'Mangle', *et cetera*).

General Firewall Information

Description

Connection Tracking

Connection tracking refers to the ability to maintain the state information about connections, such as source and destination IP address and ports pairs, connection states, protocol types and timeouts. Firewalls that do connection tracking are known as "stateful" and are inherently more secure that those who do only simple "stateless" packet processing.

The *state* of a particular connection could be **estabilished** meaning that the packet is part of already known connection, **new** meaning that the packet starts a new connection or belongs to a connection that has not seen packets in both directions yet, **related** meaning that the packet starts a new connection, but is associated with an existing connection, such as FTP data transfer or ICMP error message and, finally, **invalid** meaning that the packet does not belong to any known connection.

Connection tracking is done either in the **prerouting** chain, or the **output** chain for locally generated packets.

Another function of connection tracking which cannot be overestimated is that it is needed for NAT. You should be aware that no NAT can be performed unless you have connection tracking enabled, the same applies for p2p protocols recognition. Connection tracking also assembles IP packets from fragments before further processing.

The maximum number of connections the **/ip firewall connection** state table can contain is determined initially by the amount of physical memory present in the router. Thus, for example, a router with 64 MB of RAM can hold the information about up to 65536 connections, but a router with 128 MB RAM increases this value to more than 130000.

Please ensure that your router is equipped with sufficient amount of physical memory to properly handle all connections.

ICMP TYPE:CODE values

In order to protect your router and attached private networks, you need to configure firewall to drop or reject most of ICMP traffic. However, some ICMP packets are vital to maintain network

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

reliability or provide troubleshooting services.

The following is a list of ICMP TYPE:CODE values found in good packets. It is generally suggested to allow these types of ICMP traffic.

- • **8:0** - echo request
  - **0:0** - echo reply

    Ping

- • **11:0** - TTL exceeded
  - **3:3** - Port unreachable

    Trace

- • **3:4** - Fragmentation-DF-Set

    Path MTU discovery

General suggestion to apply ICMP filtering

- Allow ping—ICMP Echo-Request outbound and Echo-Reply messages inbound
- Allow traceroute—TTL-Exceeded and Port-Unreachable messages inbound
- Allow path MTU—ICMP Fragmentation-DF-Set messages inbound
- Block everything else

Type of Service

Internet paths vary in quality of service they provide. They can differ in cost, reliability, delay and throughput. This situation imposes some tradeoffs, *exempli gratia* the path with the lowest delay may be among the ones with the smallest throughput. Therefore, the "optimal" path for a packet to follow through the Internet may depend on the needs of the application and its user.

As the network itself has no knowledge on how to optimize path choosing for a particular application or user, the IP protocol provides a method for upper layer protocols to convey hints to the Internet Layer about how the tradeoffs should be made for the particular packet. This method is implemented with the help of a special field in the IP protocol header, the "Type of Service" field.

The fundamental rule is that if a host makes appropriate use of the TOS facility, its network service should be at least as good as it would have been if the host had not used this facility.

Type of Service (ToS) is a standard field of IP packet and it is used by many network applications and hardware to specify how the traffic should be treated by the gateway.

MikroTik RouterOS works with the full ToS byte. It does not take account of reserverd bits in this byte (because they have been redefined many times and this approach provides more flexibility). It means that it is possible to work with DiffServ marks (Differentiated Services Codepoint, DSCP as defined in RFC2474) and ECN codepoints (Explicit Congestion Notification, ECN as defined in RFC3168), which are using the same field in the IP protocol header. Note that it does not mean that RouterOS supports DiffServ or ECN, it is just possible to access and change the marks used by these protocols.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

RFC1349 defines these standard values:

- **normal** - normal service (ToS=0)
- **low-cost** - minimize monetary cost (ToS=2)
- **max-reliability** - maximize reliability (ToS=4)
- **max-throughput** - maximize throughput (ToS=8)
- **low-delay** - minimize delay (ToS=16)

Peer-to-Peer protocol filtering

Peer-to-peer protocols also known as *p2p* provide means for direct distributed data transfer between individual network hosts. While this technology powers many brilliant applications (like Skype), it is widely abused for unlicensed software and media destribution. Even when it is used for legal purposes, p2p may heavily disturb other network traffic, such as http and e-mail. RouterOS is able to recognize connections of the most popular P2P protocols and filter or enforce QOS on them.

The protocols which can be detected, are:

- **Fasttrack** (Kazaa, KazaaLite, Diet Kazaa, Grokster, iMesh, giFT, Poisoned, mlMac)
- **Gnutella** (Shareaza, XoLoX, , Gnucleus, BearShare, LimeWire (java), Morpheus, Phex, Swapper, Gtk-Gnutella (linux), Mutella (linux), Qtella (linux), MLDonkey, Acquisition (Mac OS), Poisoned, Swapper, Shareaza, XoloX, mlMac)
- **Gnutella2** (Shareaza, MLDonkey, Gnucleus, Morpheus, Adagio, mlMac)
- **DirectConnect** (DirectConnect (AKA DC++), MLDonkey, NeoModus Direct Connect, BCDC++, CZDC++)
- **eDonkey** (eDonkey2000, eMule, xMule (linux), Shareaza, MLDonkey, mlMac, Overnet)
- **Soulseek** (Soulseek, MLDonkey)
- **BitTorrent** (BitTorrent, BitTorrent++, Shareaza, MLDonkey, ABC, Azureus, BitAnarch, SimpleBT, BitTorrent.Net, mlMac)
- **Blubster** (Blubster, Piolet)
- **WPNP** (WinMX)
- **Warez** (Warez, Ares; starting from 2.8.18) - this protocol can only be dropped, speed limiting is impossible

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

DHCP Client and Server

Document revision 2.7 (Mon Apr 18 22:24:18 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
- Summary
- Quick Setup Guide
- Specifications
- Description
- Additional Documents

DHCP Client Setup
- Description
- Property Description
- Command Description
- Notes
- Example

DHCP Server Setup
- Description
- Property Description
- Notes
- Example

Store Leases on Disk
- Description
- Property Description

DHCP Networks
- Property Description
- Notes

DHCP Server Leases
- Description
- Property Description
- Command Description
- Notes
- Example

DHCP Alert
- Description
- Property Description
- Notes

DHCP Option
- Description
- Property Description
- Notes
- Example

DHCP Relay
- Description
- Property Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Notes
Example
Question&Answer-Based Setup
Command Description
Notes
Example
Dynamic Addressing, using DHCP-Relay
IP Address assignment, using FreeRADIUS Server

General Information

Summary

The DHCP (Dynamic Host Configuration Protocol) is needed for easy distribution of IP addresses in a network. The MikroTik RouterOS implementation includes both - server and client parts and is compliant with RFC2131.

General usage of DHCP:

- IP assignment in LAN, cable-modem, and wireless systems
- Obtaining IP settings on cable-modem systems

IP addresses can be bound to MAC addresses using static lease feature.

DHCP server can be used with MikroTik RouterOS HotSpot feature to authenticate and account DHCP clients. See the ***HotSpot Manual*** for more information.

Quick Setup Guide

This example will show you how to setup DHCP-Server and DHCP-Client on MikroTik RouterOS.

- Setup of a DHCP-Server.
    1. Create an IP address pool

```
/ip pool add name=dhcp-pool ranges=172.16.0.10-172.16.0.20
```

    2. Add a DHCP network which will concern to the network **172.16.0.0/12** and will distribute a gateway with IP address **172.16.0.1** to DHCP clients:

```
/ip dhcp-server network add address=172.16.0.0/12 gateway=172.16.0.1
```

    3. Finally, add a DHCP server:

```
/ip dhcp-server add interface=wlan1 address-pool=dhcp-pool
```

- Setup of the DHCP-Client (which will get a lease from the DHCP server, configured above).
    1. Add the DHCP client:

```
/ip dhcp-client add interface=wlan1 use-peer-dns=yes \
     add-default-route=yes disabled=no
```

    2. Check whether you have obtained a lease:

```
[admin@Server] ip dhcp-client> print detail
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
Flags: X - disabled, I - invalid
 0   interface=wlan1 add-default-route=yes use-peer-dns=yes status=bound
     address=172.16.0.20/12 gateway=172.16.0.1 dhcp-server=192.168.0.1
     primary-dns=159.148.147.194 expires-after=2d23:58:52
[admin@Server] ip dhcp-client>
```

Specifications

Packages required: ***dhcp***
License required: ***level1***
Home menu level: ***/ip dhcp-client, /ip dhcp-server, /ip dhcp-relay***
Standards and Technologies: ***DHCP***

Description

The DHCP protocol gives and allocates IP addresses to IP clients. DHCP is basically insecure and should only be used in trusted networks. DHCP server always listens on UDP 67 port, DHCP client - on UDP 68 port. The initial negotiation involves communication between broadcast addresses (on some phases sender will use source address of **0.0.0.0** and/or destination address of **255.255.255.255**). You should be aware of this when building firewall.

Additional Documents

- ***ISC Dynamic Host Configuration Protocol (DHCP)***
- ***DHCP mini-HOWTO***
- ***ISC DHCP FAQ***

DHCP Client Setup

Home menu level: ***/ip dhcp-client***

Description

The MikroTik RouterOS DHCP client may be enabled on any Ethernet-like interface at a time. The client will accept an address, netmask, default gateway, and two dns server addresses. The received IP address will be added to the interface with the respective netmask. The default gateway will be added to the routing table as a dynamic entry. Should the DHCP client be disabled or not renew an address, the dynamic default route will be removed. If there is already a default route installed prior the DHCP client obtains one, the route obtained by the DHCP client would be shown as invalid.

Property Description

address (*IP address* | *netmask*) - IP address and netmask, which is assigned to DHCP Client from the Server

add-default-route (yes | no; default: **yes**) - whether to add the default route to the gateway specified by the DHCP server

client-id (*text*) - corresponds to the settings suggested by the network administrator or ISP. Commonly it is set to the client's MAC address, but it may as well be any test string

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

dhcp-server (*IP address*) - IP address of the DHCP Server

enabled (yes | no; default: **no**) - whether the DHCP client is enabled

expires-after (*time*) - time, which is assigned by the DHCP Server, after which the lease expires

gateway (*IP address*) - IP address of the gateway which is assigned by DHCP Server

host-name (*text*) - the host name of the client as sent to a DHCP server

interface (*name*) - any Ethernet-like interface (this includes wireless and EoIP tunnels) on which the DHCP Client searches the DHCP Server

primary-dns (*IP address*) - IP address of the primary DNS server, assigned by the DHCP Server

secondary-dns (*IP address*) - IP address of the secondary DNS server, assigned by DHCP Server

primary-ntp - IP address of the primary NTP server, assigned by the DHCP Server

secondary-ntp - IP address of the secondary NTP server, assigned by the DHCP Server

status (*bound* | *error* | *rebinding...* | *renewing...* | *requesting...* | *searching...* | *stopped*) - shows the status of DHCP Client

use-peer-dns (yes | no; default: **yes**) - whether to accept the DNS settings advertized by DHCP server (they will be ovverriden in /ip dns submenu)

use-peer-ntp (yes | no; default: **yes**) - whether to accept the NTP settings advertized by DHCP server (they will override the settings put in the /system ntp client submenu)

Command Description

release - release current binding and restart DHCP client

renew - renew current leases. If the renew operation was not successful, client tries to reinitialize lease (i.e. it starts lease request procedure (rebind) as if it had not received an IP address yet)

Notes

If **host-name** property is not specified, client's system identity will be sent in the respective field of DHCP request.

If **client-id** property is not specified, client's MAC address will be sent in the respective field of DHCP request.

If **use-peer-dns** property is enabled, the DHCP client will unconditionally rewrite the settings in **/ip dns** submenu. In case two or more DNS servers were received, first two of them are set as primary and secondary servers respectively. In case one DNS server was received, it is put as primary server, and the secondary server is left intact.

Example

To add a DHCP client on **ether1** interface:

```
/ip dhcp-client add interface=ether1 disabled=no
[admin@MikroTik] ip dhcp-client> print detail
Flags: X - disabled, I - invalid
 0   interface=ether1 add-default-route=no use-peer-dns=no status=bound
     address=192.168.25.100/24 dhcp-server=10.10.10.1 expires-after=2d21:25:12
[admin@MikroTik] ip dhcp-client>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

DHCP Server Setup

Home menu level: ***/ip dhcp-server***

Description

The router supports an individual server for each Ethernet-like interface. The MikroTik RouterOS DHCP server supports the basic functions of giving each requesting client an IP address/netmask lease, default gateway, domain name, DNS-server(s) and WINS-server(s) (for Windows clients) information (set up in the DHCP networks submenu)

In order DHCP server to work, you must set up also IP pools (do not include the DHCP server's IP address into the pool range) and DHCP networks.

It is also possible to hand out leases for DHCP clients using the RADIUS server, here are listed the parameters for used in RADIUS server.

Access-Request:

- **NAS-Identifier** - router identity
- **NAS-IP-Address** - IP address of the router itself
- **NAS-Port** - unique session ID
- **NAS-Port-Type** - Ethernet
- **Calling-Station-Id** - client identifier (active-client-id)
- **Framed-IP-Address** - IP address of the client (active-address)
- **Called-Station-Id** - name of DHCP server
- **User-Name** - MAC address of the client (active-mac-address)
- **Password** - ""

Access-Accept:

- **Framed-IP-Address** - IP address that will be assigned to client
- **Framed-Pool** - ip pool from which to assign ip address to client
- **Rate-Limit** - Datarate limitation for clients DHCP clients. Format is: rx-rate[/tx-rate] [rx-burst-rate[/tx-burst-rate] [rx-burst-threshold[/tx-burst-threshold] [rx-burst-time[/tx-burst-time]]]]. All rates should be numbers with optional 'k' (1,000s) or 'M' (1,000,000s). If tx-rate is not specified, rx-rate is as tx-rate too. Same goes for tx-burst-rate and tx-burst-threshold and tx-burst-time. If both rx-burst-threshold and tx-burst-threshold are not specified (but burst-rate is specified), rx-rate and tx-rate is used as burst thresholds. If both rx-burst-time and tx-burst-time are not specified, 1s is used as default.
- **Ascend-Data-Rate** - tx/rx data rate limitation if multiple attributes are provided, first limits tx data rate, second - rx data rate. If used together with Ascend-Xmit-Rate, specifies rx rate. 0 if unlimited
- **Ascend-Xmit-Rate** - tx data rate limitation. It may be used to specify tx limit only instead of sending two sequental Ascend-Data-Rate attributes (in that case Ascend-Data-Rate will specify the receive rate). 0 if unlimited
- **Session-Timeout** - max lease time (lease-time)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Property Description

add-arp (yes | no; default: **no**) - whether to add dynamic ARP entry:

- **no** - either ARP mode should be enabled on that interface or static ARP entries should be administratively defined in /ip arp submenu

address-pool (*name* | *static-only*; default: **static-only**) - IP pool, from which to take IP addresses for clients

- **static-only** - allow only the clients that have a static lease (i.e. no dynamic addresses will be given to clients, only the ones added in lease submenu)

always-broadcast (yes | no; default: **no**) - always send replies as broadcasts

authoritative (yes | no; default: **no**) - whether the DHCP server is the only one DHCP server for that network

bootp-support (*none* | *static* | *dynamic*; default: **static**) - support for BOOTP clients

- **none** - do not respond to BOOTP requests
- **static** - offer only static leases to BOOTP clients
- **dynamic** - offer static and dynamic leases for BOOTP clients

delay-threshold (*time*; default: **none**) - if secs field in DHCP packet is smaller than delay-threshold, then this packet is ignored

- **none** - there is no threshold (all DHCP packets are processed)

interface (*name*) - Ethernet-like interface name

lease-time (*time*; default: **72h**) - the time that a client may use an address. The client will try to renew this address after a half of this time and will request a new address after time limit expires

name (*name*) - reference name

ntp-server (*text*) - the DHCP client will use these as the default NTP servers. Two comma-separated NTP servers can be specified to be used by DHCP client as primary and secondary NTP servers

relay (*IP address*; default: **0.0.0.0**) - the IP address of the relay this DHCP server should process requests from:

- **0.0.0.0** - the DHCP server will be used only for direct requests from clients (no DHCP really allowed)
- **255.255.255.255** - the DHCP server should be used for any incomming request from a DHCP relay except for those, which are processed by another DHCP server that exists in the /ip dhcp-server submenu

src-address (*IP address*; default: **0.0.0.0**) - the address which the DHCP client must send requests to in order to renew an IP address lease. If there is only one static address on the DHCP server interface and the source-address is left as 0.0.0.0, then the static address will be used. If there are multiple addresses on the interface, an address in the same subnet as the range of given addresses should be used

use-radius (yes | no; default: **no**) - whether to use RADIUS server for dynamic leases

Notes

If using both - Universal Client and DHCP Server on the same interface, client will only receive a

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

DHCP lease in case it is directly reachable by its MAC address through that interface (some wireless bridges may change client's MAC address).

If **authoritative** property is set to **yes**, the DHCP server is sending rejects for the leases it cannot bind or renew. It also may (although not always) help to prevent the users of the network to run illicitly their own DHCP servers disturbing the proper way this network should be functioning.

If **relay** property of a DHCP server is not set to **0.0.0.0** the DHCP server will not respond to the direct requests from clients.

Example

To add a DHCP server to interface **ether1**, lending IP addresses from **dhcp-clients** IP pool for 2 hours:

```
/ip dhcp-server add name=dhcp-office disabled=no address-pool=dhcp-clients \
interface=ether1 lease-time=2h
[admin@MikroTik] ip dhcp-server> print
Flags: X - disabled, I - invalid
 #   NAME            INTERFACE RELAY          ADDRESS-POOL LEASE-TIME ADD-ARP
 0   dhcp-office     ether1                   dhcp-clients 02:00:00
[admin@MikroTik] ip dhcp-server>
```

Store Leases on Disk

Home menu level: ***/ip dhcp-server config***

Description

Leases are always stored on disk on graceful shutdown and reboot. If on every lease change it is stored on disk, a lot of disk writes happen. There are no problems if it happens on a hard drive, but is very bad on Compact Flash (especially, if lease times are very short). To minimize writes on disk, all changes are flushed together every **store-leases-disk** seconds. If this time will be very short (immediately), then no changes will be lost even in case of hard reboots and power losts. But, on CF there may be too many writes in case of short lease times (as in case of hotspot). If this time will be very long (never), then there will be no writes on disk, but information about active leases may be lost in case of power loss. In these cases dhcp server may give out the same ip address to another client, if first one will not respond to ping requests.

Property Description

store-leases-disk (*time-interval* | *immediately* | *never*; default: **5min**) - how frequently lease changes should be stored on disk

DHCP Networks

Home menu level: ***/ip dhcp-server network***

Property Description

address (*IP address* | *netmask*) - the network DHCP server(s) will lend addresses from

boot-file-name (*text*) - Boot file name

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

dhcp-option (*text*) - add additional DHCP options from /ip dhcp-server option list. You cannot redefine parameters which are already defined in this submenu:

- **Subnet-Mask (code 1)** - netmask
- **Router (code 3)** - gateway
- **Domain-Server (code 6)** - dns-server
- **Domain-Name (code 15)** - domain
- **NETBIOS-Name-Server** - wins-server

dns-server (*text*) - the DHCP client will use these as the default DNS servers. Two comma-separated DNS servers can be specified to be used by DHCP client as primary and secondary DNS servers

domain (*text*) - the DHCP client will use this as the 'DNS domain' setting for the network adapter

gateway (*IP address*; default: **0.0.0.0**) - the default gateway to be used by DHCP clients

netmask (*integer*: 0..32; default: **0**) - the actual network mask to be used by DHCP client

- **0** - netmask from network address is to be used

next-server (*IP address*) - IP address of next server to use in bootstrap

wins-server (*text*) - the Windows DHCP client will use these as the default WINS servers. Two comma-separated WINS servers can be specified to be used by DHCP client as primary and secondary WINS servers

Notes

The **address** field uses netmask to specify the range of addresses the given entry is valid for. The actual netmask clients will be using is specified in **netmask** property.

DHCP Server Leases

Home menu level: ***/ip dhcp-server lease***

Description

DHCP server lease submenu is used to monitor and manage server's leases. The issued leases are showed here as dynamic entries. You can also add static leases to issue the definite client (determined by MAC address) the specified IP address.

Generally, the DHCP lease it allocated as follows:

1. an unused lease is in **waiting** state
2. if a client asks for an IP address, the server chooses one
3. if the client will receive statically assigned address, the lease becomes **offered**, and then **bound** with the respective lease time
4. if the client will receive a dynamic address (taken from an IP address pool), the router sends a ping packet and waits for answer for 0.5 seconds. During this time, the lease is marked **testing**
5. in case, the address does not respond, the lease becomes **offered**, and then **bound** with the respective lease time

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

6. in other case, the lease becomes **busy** for the lease time (there is a command to retest all busy addresses), and the client's request remains unanswered (the client will try again shortly)

A client may free the leased address. When the dynamic lease is removed, and the allocated address is returned to the address pool. But the static lease becomes **busy** until the client will reacquire the address.

Note that the IP addresses assigned statically are not probed.

Property Description

active-address (*read-only: IP address*) - actual IP address for this lease

active-client-id (*read-only: text*) - actual client-id of the client

active-mac-address (*read-only: MAC address*) - actual MAC address of the client

active-server (*read-only:*) - actual dhcp server, which serves this client

address (*IP address*) - specify ip address (or ip pool) for static lease

- **0.0.0.0** - use pool from server

agent-circuit-id (*read-only: text*) - circuit ID of DHCP relay agent

agent-remote-id (*read-only: text*) - Remote ID, set by DHCP relay agent

block-access (yes | no; default: **no**) - block access for this client (drop packets from this client)

client-id (*text*; default: **""**) - if specified, must match DHCP 'client identifier' option of the request

expires-after (*read-only: time*) - time until lease expires

host-name (*read-only: text*) - shows host name option from last received DHCP request

lease-time (*time*; default: **0s**) - time that the client may use an address

- **0s** - lease will never expire

mac-address (*MAC address*; default: **00:00:00:00:00:00**) - if specified, must match MAC address of the client

radius (*read-only:* yes | no) - shows, whether this dynamic lease is authenticated by RADIUS or not

rate-limit (*read-only: text*; default: **""**) - sets rate limit for active lease. Format is: rx-rate[/tx-rate] [rx-burst-rate[/tx-burst-rate] [rx-burst-threshold[/tx-burst-threshold] [rx-burst-time[/tx-burst-time]]]]. All rates should be numbers with optional 'k' (1,000s) or 'M' (1,000,000s). If tx-rate is not specified, rx-rate is as tx-rate too. Same goes for tx-burst-rate and tx-burst-threshold and tx-burst-time. If both rx-burst-threshold and tx-burst-threshold are not specified (but burst-rate is specified), rx-rate and tx-rate is used as burst thresholds. If both rx-burst-time and tx-burst-time are not specified, 1s is used as default.

rx-rate (*integer*; default: **0**) - maximal receive bitrate to the client (for users it is upload bitrate))

- **0** - no limitation

server (*read-only: name*) - server name which serves this client

status (*read-only: waiting* | *testing* | *authorizing* | *busy* | *offered* | *bound*) - lease status:

- **waiting** - not used static lease
- **testing** - testing whether this address is used or not (only for dynamic leases) by pinging it with timeout of 0.5s

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **authorizing** - waiting for response from radius server
- **busy** - this address is assigned statically to a client or already exists in the network, so it can not be leased
- **offered** - server has offered this lease to a client, but did not receive confirmation from the client
- **bound** - server has received client's confirmation that it accepts offered address, it is using it now and will free the address not later, than the lease time will be over

tx-rate (*integer*; default: **0**) - maximal transmit bitrate to the client (for users it is download bitrate))

- **0** - no limitation

Command Description

check-status - Check status of a given busy dynamic lease, and free it in case of no response

make-static - convert a dynamic lease to static one

Notes

If **rate-limit** is specified, a simple queue is added with corresponding parameters when lease enters bound state. Arp entry is added right after adding of queue is done (only if add-arp is enabled for dhcp server). To be sure, that client cannot use his ip address without getting dhcp lease and thus avoiding rate-limit, reply-only mode must be used on that ethernet interface.

Even though client address may be changed (with adding a new item) in **lease print** list, it will not change for the client. It is true for any changes in the DHCP server configuration because of the nature of the DHCP protocol. Client tries to renew assigned IP address only when half a lease time is past (it tries to renew several times). Only when full lease time is past and IP address was not renewed, new lease is asked (rebind operation).

the deault **mac-address** value will never work! You should specify a correct MAC address there.

Example

To assign 10.5.2.100 static IP address for the existing DHCP client (shown in the lease table as item #0):

```
[admin@MikroTik] ip dhcp-server lease> print
Flags: X - disabled, H - hotspot, D - dynamic
  #   ADDRESS         MAC-ADDRESS       EXPIRES-AFTER SERVER         STATUS
  0 D 10.5.2.90       00:04:EA:C6:0E:40 1h48m59s      switch         bound
  1 D 10.5.2.91       00:04:EA:99:63:C0 1h42m51s      switch         bound
[admin@MikroTik] ip dhcp-server lease> add copy-from=0 address=10.5.2.100
[admin@MikroTik] ip dhcp-server lease> print
Flags: X - disabled, H - hotspot, D - dynamic
  #   ADDRESS         MAC-ADDRESS       EXPIRES-AFTER SERVER         STATUS
  1 D 10.5.2.91       00:04:EA:99:63:C0 1h42m18s      switch         bound
  2   10.5.2.100      00:04:EA:C6:0E:40 1h48m26s      switch         bound
[admin@MikroTik] ip dhcp-server lease>
```

DHCP Alert

Home menu level: ***/ip dhcp-server alert***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

To find any rogue DHCP servers as soon as they appear in your network, DHCP Alert tool can be used. It will monitor ethernet for all DHCP replies and check, whether this reply comes from a valid DHCP server. If reply from unknown DHCP server is detected, alert gets triggered:

```
[admin@MikroTik] ip dhcp-server alert>/log print
00:34:23 dhcp,critical,error,warning,info,debug dhcp alert on Public:
    discovered unknown dhcp server, mac 00:02:29:60:36:E7, ip 10.5.8.236
[admin@MikroTik] ip dhcp-server alert>
```

When the system alerts about a rogue DHCP server, it can execute a custom script.

As DHCP replies can be unicast, rogue dhcp detector may not receive any offer to other dhcp clients at all. To deal with this, rogue dhcp server acts as a dhcp client as well - it sends out dhcp discover requests once a minute

Property Description

alert-timeout (*none* | *time*; default: **none**) - time, after which alert will be forgotten. If after that time the same server will be detected, new alert will be generated

- **none** - infinite time

interface (*name*) - interface, on which to run rogue DHCP server finder

invalid-server (*read-only: text*) - list of MAC addresses of detected unknown DHCP servers. Server is removed from this list after alert-timeout

on-alert (*text*) - script to run, when an unknown DHCP server is detected

valid-server (*text*) - list of MAC addresses of valid DHCP servers

Notes

All alerts on an interface can be cleared at any time using command: **/ip dhcp-server alert reset-alert <interface>**

Note, that e-mail can be sent, using */system logging action add target=email*

DHCP Option

Home menu level: ***/ip dhcp-server option***

Description

With help of DHCP Option, it is possible to define additional custom options for DHCP Server.

Property Description

code (*integer*: 1..254) - dhcp option code. All codes are available at http://www.iana.org/assignments/bootp-dhcp-parameters

name (*name*) - descriptive name of the option

value (*text*) - parameter's value in form of a string. If the string begins with "0x", it is assumed as a

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

hexadecimal value

Notes

The defined options you can use in */ip dhcp-server network* submenu

According to the DHCP protocol, a parameter is returned to the DHCP client only if it requests this parameter, specifying the respective code in DHCP request Parameter-List (code 55) attribute. If the code is not included in Parameter-List attribute, DHCP server will not send it to the DHCP client.

Example

This example shows how to set DHCP server to reply on DHCP client's Hostname request (code 12) with value **Host-A**.

Add an option named **Option-Hostname** with code **12** (Hostname) and value **Host-A**:

```
[admin@MikroTik] ip dhcp-server option> add name=Hostname code=12 \
value="Host-A"
[admin@MikroTik] ip dhcp-server option> print
 # NAME                                CODE VALUE
 0 Option-Hostname                     12   Host-A
[admin@MikroTik] ip dhcp-server option>
```

Use this option in DHCP server network list:

```
[admin@MikroTik] ip dhcp-server network> add address=10.1.0.0/24 \
\... gateway=10.1.0.1 dhcp-option=Option-Hostname dns-server=159.148.60.20
[admin@MikroTik] ip dhcp-server network> print detail
 0 address=10.1.0.0/24 gateway=10.1.0.1 dns-server=159.148.60.20
   dhcp-option=Option-Hostname
[admin@MikroTik] ip dhcp-server network>
```

Now the DHCP server will reply with its Hostname **Host-A** to DHCP client (if requested)

DHCP Relay

Home menu level: ***/ip dhcp-relay***

Description

DHCP Relay is just a proxy that is able to receive a DHCP request and resend it to the real DHCP server

Property Description

dhcp-server (*text*) - list of DHCP servers' IP addresses which should the DHCP requests be forwarded to

delay-threshold (*time*; default: **none**) - if secs field in DHCP packet is smaller than delay-threshold, then this packet is ignored

interface (*name*) - interface name the DHCP relay will be working on

local-address (*IP address*; default: **0.0.0.0**) - the unique IP address of this DHCP relay needed for DHCP server to distinguish relays:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **0.0.0.0** - the IP address will be chosen automatically

name (*name*) - descriptive name for relay

Notes

DHCP relay does not choose the particular DHCP server in the dhcp-server list, it just sent to all the listed servers.

Example

To add a DHCP relay named **relay** on **ether1** interface resending all received requests to the **10.0.0.1** DHCP server:

```
[admin@MikroTik] ip dhcp-relay> add name=relay interface=ether1 \
\... dhcp-server=10.0.0.1 disabled=no
[admin@MikroTik] ip dhcp-relay> print
Flags: X - disabled, I - invalid
  #   NAME                            INTERFACE DHCP-SERVER     LOCAL-ADDRESS
  0   relay                           ether1    10.0.0.1        0.0.0.0

[admin@MikroTik] ip dhcp-relay>
```

Question&Answer-Based Setup

Command name: ***/ip dhcp-server setup***

Command Description

addresses to give out (*text*) - the pool of IP addresses DHCP server should lease to the clients

dhcp address space (*IP address* | *netmask*; default: **192.168.0.0/24**) - network the DHCP server will lease to the clients

dhcp relay (*IP address*; default: **0.0.0.0**) - the IP address of the DHCP relay between the DHCP server and the DHCP clients

dhcp server interface (*name*) - interface to run DHCP server on

dns servers (*IP address*) - IP address of the appropriate DNS server to be propagated to the DHCP clients

gateway (*IP address*; default: **0.0.0.0**) - the default gateway of the leased network

lease time (*time*; default: **3d**) - the time the lease will be valid

Notes

Depending on current settings and answers to the previous questions, default values of following questions may be different. Some questions may disappear if they become redundant (for example, there is no use of asking for 'relay' when the server will lend the directly connected network)

Example

To configure DHCP server on **ether1** interface to lend addresses from 10.0.0.2 to 10.0.0.254 which

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

belong to the **10.0.0.0/24** network with **10.0.0.1** gateway and **159.148.60.2** DNS server for the time of 3 days:

```
[admin@MikroTik] ip dhcp-server> setup
Select interface to run DHCP server on

dhcp server interface: ether1
Select network for DHCP addresses

dhcp address space: 10.0.0.0/24
Select gateway for given network

gateway for dhcp network: 10.0.0.1
Select pool of ip addresses given out by DHCP server

addresses to give out: 10.0.0.2-10.0.0.254
Select DNS servers

dns servers: 159.148.60.20
Select lease time

lease time: 3d
[admin@MikroTik] ip dhcp-server>
```

The wizard has made the following configuration based on the answers above:

```
[admin@MikroTik] ip dhcp-server> print
Flags: X - disabled, I - invalid
  #   NAME            INTERFACE RELAY           ADDRESS-POOL LEASE-TIME ADD-ARP
  0   dhcp1           ether1    0.0.0.0         dhcp_pool1   3d         no

[admin@MikroTik] ip dhcp-server> network print
  # ADDRESS            GATEWAY         DNS-SERVER      WINS-SERVER     DOMAIN
  0 10.0.0.0/24        10.0.0.1        159.148.60.20

[admin@MikroTik] ip dhcp-server> /ip pool print
  # NAME                                      RANGES
  0 dhcp_pool1                                10.0.0.2-10.0.0.254

[admin@MikroTik] ip dhcp-server>
```

General Information

Dynamic Addressing, using DHCP-Relay

Let us consider that you have several IP networks 'behind' other routers, but you want to keep all DHCP servers on a single router. To do this, you need a DHCP relay on your network which relies DHCP requests from clients to DHCP server.

This example will show you how to configure a DHCP server and a DHCP relay which serve 2 IP networks - **192.168.1.0/24** and **192.168.2.0/24** that are behind a router **DHCP-Relay**.

IP addresses of **DHCP-Server**:

```
[admin@DHCP-Server] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   192.168.0.1/24     192.168.0.0     192.168.0.255   To-DHCP-Relay
 1   10.1.0.2/24 10.1.0.0 10.1.0.255 Public
[admin@DHCP-Server] ip address>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

IP addresses of **DHCP-Relay**:

```
[admin@DHCP-Relay] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   192.168.0.1/24     192.168.0.0     192.168.0.255   To-DHCP-Server
 1   192.168.1.1/24     192.168.1.0     192.168.1.255   Local1
 2   192.168.2.1/24     192.168.2.0     192.168.2.255   Local2
[admin@DHCP-Relay] ip address>
```

To setup 2 DHCP Servers on **DHCP-Server** router add 2 pools. For networks **192.168.1.0/24** and **192.168.2.0**:

```
/ip pool add name=Local1-Pool ranges=192.168.1.11-192.168.1.100
/ip pool add name=Local1-Pool ranges=192.168.2.11-192.168.2.100

[admin@DHCP-Server] ip pool> print
 # NAME                                      RANGES
 0 Local1-Pool                               192.168.1.11-192.168.1.100
 1 Local2-Pool                               192.168.2.11-192.168.2.100
[admin@DHCP-Server] ip pool>
```

Create DHCP Servers:

```
/ip dhcp-server add interface=To-DHCP-Relay relay=192.168.1.1 \
   address-pool=Local1-Pool name=DHCP-1 disabled=no
/ip dhcp-server add interface=To-DHCP-Relay relay=192.168.2.1 \
   address-pool=Local2-Pool name=DHCP-2 disabled=no

[admin@DHCP-Server] ip dhcp-server> print
Flags: X - disabled, I - invalid
 #   NAME         INTERFACE     RELAY           ADDRESS-POOL LEASE-TIME ADD-ARP
 0   DHCP-1       To-DHCP-Relay 192.168.1.1     Local1-Pool  3d00:00:00
 1   DHCP-2       To-DHCP-Relay 192.168.2.1     Local2-Pool  3d00:00:00
[admin@DHCP-Server] ip dhcp-server>
```

Configure respective networks:

```
/ip dhcp-server network add address=192.168.1.0/24 gateway=192.168.1.1 \
   dns-server=159.148.60.20
/ip dhcp-server network add address=192.168.2.0/24 gateway=192.168.2.1 \
   dns-server 159.148.60.20

[admin@DHCP-Server] ip dhcp-server network> print
 # ADDRESS            GATEWAY         DNS-SERVER      WINS-SERVER     DOMAIN
 0 192.168.1.0/24     192.168.1.1     159.148.60.20
 1 192.168.2.0/24     192.168.2.1     159.148.60.20
[admin@DHCP-Server] ip dhcp-server network>
```

Configuration of **DHCP-Server** is done. Now let's configure **DHCP-Relay**:

```
/ip dhcp-relay add name=Local1-Relay interface=Local1 \
   dhcp-server=192.168.0.1 local-address=192.168.1.1 disabled=no
/ip dhcp-relay add name=Local2-Relay interface=Local2 \
   dhcp-server=192.168.0.1 local-address=192.168.2.1 disabled=no

[admin@DHCP-Relay] ip dhcp-relay> print
Flags: X - disabled, I - invalid
 #   NAME                      INTERFACE       DHCP-SERVER     LOCAL-ADDRESS
 0   Local1-Relay              Local1          192.168.0.1     192.168.1.1
 1   Local2-Relay              Local2          192.168.0.1     192.168.2.1
[admin@DHCP-Relay] ip dhcp-relay>
```

IP Address assignment, using FreeRADIUS Server

Let us consider that we want to assign IP addresses for clients, using the RADIUS server.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

We assume that you already have installed FreeRADIUS. Just add these lines to specified files:

users file:

```
00:0B:6B:31:02:4B       Auth-Type := Local, Password == ""
        Framed-IP-Address = 192.168.0.55
```

clients.conf file

```
client 172.16.0.1 {
    secret = MySecret
    shortname = Server
}
```

Configure Radius Client on RouterOS:

```
/radius add service=dhcp address=172.16.0.2 secret=MySecret

[admin@DHCP-Server] radius> print detail
Flags: X - disabled
 0   service=dhcp called-id="" domain="" address=172.16.0.2 secret="MySecret"
     authentication-port=1812 accounting-port=1813 timeout=00:00:00.300
     accounting-backup=no realm=""
[admin@DHCP-Server] radius>
```

Setup DHCP Server:

1. Create an address pool:

```
/ip pool add name=Radius-Clients ranges=192.168.0.11-192.168.0.100
```

2. Add a DHCP server:

```
/ip dhcp-server add address-pool=Radius-Clients use-radius=yes interface=Local \
   disabled=no
```

3. Configure DHCP networks:

```
/ip dhcp-server network add address=192.168.0.0/24 gateway=192.168.0.1 \
dns-server=159.148.147.194,159.148.60.20
```

Now the client with MAC address **00:0B:6B:31:02:4B** will always receive IP address **192.168.0.55**.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

DNS Client and Cache

Document revision 1.2 (Fri Apr 15 17:37:43 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
  Additional Documents
Client Configuration and Cache Setup
  Description
  Property Description
  Notes
  Example
Cache Monitoring
  Property Description
Static DNS Entries
  Description
  Property Description
  Example
Flushing DNS cache
  Command Description
  Example

General Information

Summary

DNS cache is used to minimize DNS requests to an external DNS server as well as to minimize DNS resolution time. This is a simple recursive DNS server with local items.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip dns***
Standards and Technologies: ***DNS***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***HotSpot Gateway***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

•

Description

The MikroTik router with DNS cache feature enabled can be set as a primary DNS server for any DNS-compliant clients. Moreover, MikroTik router can be specified as a primary DNS server under its dhcp-server settings. When the DNS cache is enabled, the MikroTik router responds to DNS TCP and UDP requests on port 53.

Additional Documents

- ***http://www.freesoft.org/CIE/Course/Section2/3.htm***
- ***http://www.networksorcery.com/enp/protocol/dns.htm***
- ***RFC1035***

Client Configuration and Cache Setup

Home menu level: ***/ip dns***

Description

DNS client is used to provide domain name resolution for router itself as well as for the P2P clients connected to the router.

Property Description

allow-remote-requests (yes | no) - specifies whether to allow network requests

cache-max-ttl (*time*; default: **1w**) - specifies maximum time-to-live for cahce records. In other words, cache records will expire after cache-max-ttl time.

cache-size (*integer*: 512..10240; default: **2048KiB**) - specifies the size of DNS cache in KiB

cache-used (*read-only: integer*) - displays the currently used cache size in KiB

primary-dns (*IP address*; default: **0.0.0.0**) - primary DNS server

secondary-dns (*IP address*; default: **0.0.0.0**) - secondary DNS server

Notes

If the property **use-peer-dns** under **/ip dhcp-client** is set to **yes** then **primary-dns** under **/ip dns** will change to a DNS address given by DHCP Server.

Example

To set 159.148.60.2 as the primary DNS server and allow the router to be used as a DNS server, do the following:

```
[admin@MikroTik] ip dns> set primary-dns=159.148.60.2 \
\... allow-remote-requests=yes
[admin@MikroTik] ip dns> print
            primary-dns: 159.148.60.2
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
         secondary-dns: 0.0.0.0
  allow-remote-requests: yes
             cache-size: 2048KiB
          cache-max-ttl: 1w
             cache-used: 17KiB
[admin@MikroTik] ip dns>
```

Cache Monitoring

Home menu level: ***/ip dns cache***

Property Description

address (*read-only: IP address*) - IP address of the host

name (*read-only: name*) - DNS name of the host

ttl (*read-only: time*) - remaining time-to-live for the record

Static DNS Entries

Home menu level: ***/ip dns static***

Description

The MikroTik RouterOS has an embedded DNS server feature in DNS cache. It allows you to link the particular domain names with the respective IP addresses and advertize these links to the DNS clients using the router as their DNS server.

Property Description

address (*IP address*) - IP address to resolve domain name with

name (*text*) - DNS name to be resolved to a given IP address

ttl (*time*) - time-to-live of the DNS record

Example

To add a static DNS entry for **www.example.com** to be resolved to **10.0.0.1** IP address:

```
[admin@MikroTik] ip dns static> add name www.example.com address=10.0.0.1
[admin@MikroTik] ip dns static> print
 # NAME                                                ADDRESS         TTL
 0 aaa.aaa.a                                           123.123.123.123 1d
 1 www.example.com                                     10.0.0.1        1d
[admin@MikroTik] ip dns static>
```

Flushing DNS cache

Command name: ***/ip dns cache flush***

Command Description

flush - clears internal DNS cache

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

```
[admin@MikroTik] ip dns> cache flush
[admin@MikroTik] ip dns> print
              primary-dns: 159.148.60.2
            secondary-dns: 0.0.0.0
    allow-remote-requests: yes
               cache-size: 2048 KiB
            cache-max-ttl: 1w
               cache-used: 10 KiB
[admin@MikroTik] ip dns>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

HotSpot Gateway

Document revision 4 (Tue Oct 04 17:03:23 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Quick Setup Guide
  Specifications
  Description
Question&Answer-Based Setup
  Command Description
  Notes
  Example
HotSpot Interface Setup
  Description
  Property Description
  Command Description
  Notes
  Example
HotSpot Server Profiles
  Property Description
  Notes
  Example
HotSpot User Profiles
  Description
HotSpot Users
  Description
HotSpot Active Users
  Description
HotSpot Cookies
  Description
  Property Description
  Notes
  Example
HTTP-level Walled Garden
  Description
  Property Description
  Notes
  Example
IP-level Walled Garden
  Description
  Property Description
  Example
One-to-one NAT static address bindings

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description
Property Description
Notes
Active Host List
Description
Property Description
Command Description
Service Port
Description
Property Description
Example
Customizing HotSpot: Firewall Section
Description
Customizing HotSpot: HTTP Servlet Pages
Description
Notes
Example
Possible Error Messages
Description
HotSpot How-to's
Description

General Information

Summary

The MikroTik HotSpot Gateway enables providing of public network access for clients using wireless or wired network connections.

HotSpot Gateway features:

- authentication of clients using local client database, or RADIUS server
- accounting using local database, or RADIUS server
- Walled-garden system (accessing some web pages without authorization)

Quick Setup Guide

The most noticeable difference in user experience setting up HotSpot system in version 2.9 from the previous RouterOS versions is that it has become in order of magnitude easier to set up a correctly working HotSpot system.

Given a router with two interfaces: Local (where HotSpot clients are connected to) and Public, which is connected to the Internet. To set up HotSpot on the Local interface:

1. first, a valid IP config is required on both interfaces. This can be done with **/setup** command. In this example we will assume the configuration with DHCP server on the Local interface
2. valid DNS configuration must be set up in the **/ip dns** submenu
3. To put HotSpot on the Local interface, using the same IP address pool as DHCP server uses

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

for that interface: `/ip hotspot add interface=local address-pool=dhcp-pool-1`

4. and finally, add at least one HotSpot user: `/ip hotspot user add name=admin`

These simple steps should be sufficient to enable HotSpot system

Please find many HotSpot How-to's, which will answer most of your questions about configuring a HotSpot gateway, at the end of this manual. It is still recommended that you read and understand all the **Description** section below before deploying a HotSpot system.

If this does not work:

- check that **/ip dns** contains valid DNS servers, try to **/ping www.mikrotik.com** to see, that DNS resolving works
- make sure that connection tracking is enabled: `/ip firewall connection tracking set enabled=yes`

Specifications

Packages required: ***hotspot, dhcp (optional)***
License required: ***level1 (Limited to 1 active user), level3 (Limited to 1 active user), level4 (Limited to 200 active users), level5 (Limited to 500 active users), level6***
Home menu level: ***/ip hotspot***
Standards and Technologies: ***ICMP, DHCP***
Hardware usage: ***Not significant***

Description

MikroTik HotSpot Gateway should have at least two network interfaces:

1. HotSpot interface, which is used to connect HotSpot clients
2. LAN/WAN interface, which is used to access network resources. For example, DNS and RADIUS server(s) should be accessible

The diagram below shows a sample HotSpot setup.

The HotSpot interface should have an IP address assigned to it. Physical network connection has to be established between the HotSpot user's computer and the gateway. It can be wireless (the wireless card should be registered to AP), or wired (the NIC card should be connected to a hub or a switch).

Note that the most noticeable difference in user experience setting up HotSpot system in version 2.9 from the previous RouterOS versions is that it has become in order of magnitude easier to set up a correctly working HotSpot system.

Introduction to HotSpot

HotSpot is a way to authorize users to access some network resources. It does not provide traffic encryption. To log in, users may use almost any web browser (either HTTP or HTTPS protocol), so they are not required to install additional software. The gateway is accounting the uptime and amount of traffic each of its clients have used, and also can send this information to a RADIUS server. The HotSpot system may limit each particular user's bitrate, total amount of traffic, uptime

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

and some other parameters mentioned further in this document.

The HotSpot system is targeted to provide authentication within a local network (to access the Internet), but may as well be used to authorize access from outer networks to access local resources. Configuring Walled Garden feature, it is possible to allow users to access some web pages without the need of prior authentication.

Getting Address

First of all, a client must get an IP address. It may be set on the client statically, or leased from a DHCP server. The DHCP server may provide ways of binding lent IP addresses to clients MAC addresses, if required. The HotSpot system does not care how did a client get an address before he/she gets to the HotSpot login page.

Moreover, HotSpot server may automatically and transparently change any IP address (yes, meaning really **any** IP address) of a client to a valid unused address from the selected IP pool. This feature gives a possibility to provide a network access (for example, Internet access) to mobile clients that are not willing (or are disallowed, not qualified enough or otherwise unable) to change their networking settings. The users will not notice the translation (i.e., there will not be any changes in the users' config), but the router itself will see completely different (from what is actually set on each client) source IP addresses on packets sent from the clients (even firewall mangle table will 'see' the translated addresses). This technique is called one-to-one NAT, but is also known as "Universal Client" as that is how it was called in the RouterOS version 2.8.

One-to-one NAT accepts any incoming address from a connected network interface and performs a network address translation so that data may be routed through standard IP networks. Clients may use any preconfigured addresses. If the one-to-one NAT feature is set to translate a client's address to a public IP address, then the client may even run a server or any other service that requires a public IP address. This NAT is changing source address of each packet just after it is received by the router (it is like source NAT that is performed earlier, so that even firewall mangle table, which normally 'sees' received packets unaltered, can only 'see' the translated address).

Note also that **arp** mode must be **enabled** on the interface you use one-to-one NAT on.

Before the authentication

When enabling HotSpot on an interface, the system automatically sets up everything needed to show login page for all clients that are not logged in. This is done by adding dynamic destination NAT rules, which you can observe on a working HotSpot system. These rules are needed to redirect all HTTP and HTTPS requests from unauthorized users to the HotSpot servlet (i.e., the authentication procedure, e.g., the login page). Other rules that are also inserted, we will describe later in a special section of this manual.

In most common setup, opening any HTTP page will bring up the HotSpot servlet login page (which can be customized extensively, as will be described later on). As normal user behavior is to open web pages by their DNS names, a valid DNS configuration should be set up on the HotSpot gateway itself (it is possible to reconfigure the gateway so that it will not require local DNS configuration, but such a configuration is impractical and thus not recommended).

Walled Garden

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

You may wish not to require authorization for some services (for example to let clients access the web server of your company without registration), or even to require authorization only to a number of services (for example, for users to be allowed to access an internal file server or another restricted area). This can be done by setting up Walled Garden system.

When a not logged-in user requests a service allowed in the Walled Garden configuration, the HotSpot gateway does not intercept it, or in case of HTTP, simply redirects the request to the original destination (or to a specified parent proxy). When a user is logged in, there is no effect of this table on him/her.

To implement the Walled Garden feature for HTTP requests, an embedded web proxy server has been designed, so all the requests from not authorized users are really going through this proxy. **Note** that the embedded proxy server does not have caching function yet. Also note that this embedded proxy server is in the **system** software package and does not require **web-proxy** package. It is configurable under **/ip proxy**

Authentication

- **HTTP PAP** - simplest method, which shows the HotSpot login page and expect to get the authentication info (i.e. username and password) in plain text. Note that passwords are not being encrypted when transferred over the network. An another use of this method is the possibility of hard-coded authentication information in the servlet's login page simply creating the appropriate link.
- **HTTP CHAP** - standard method, which includes CHAP challenge in the login page. The CHAP MD5 hash challenge is to be used together with the user's password for computing the string which will be sent to the HotSpot gateway. The hash result (as a password) together with username is sent over network to HotSpot service (so, password is never sent in plain text over IP network). On the client side, MD5 algorithm is implemented in JavaScript applet, so if a browser does not support JavaScript (like, for example, Internet Explorer 2.0 or some PDA browsers), it will not be able to authenticate users. It is possible to allow unencrypted passwords to be accepted by turning on HTTP PAP authentication method, but it is not recommended (because of security considerations) to use that feature.
- **HTTPS** - the same as HTTP PAP, but using SSL protocol for encrypting transmissions. HotSpot user just send his/her password without additional hashing (note that there is no need to worry about plain-text password exposure over the network, as the transmission itself is encrypted). In either case, HTTP POST method (if not possible, then - HTTP GET method) is used to send data to the HotSpot gateway.
- **HTTP cookie** - after each successful login, a cookie is sent to web browser and the same cookie is added to active HTTP cookie list. Next time the same user will try to log in, web browser will send http cookie. This cookie will be compared with the one stored on the HotSpot gateway and only if source MAC address and randomly generated ID match the ones stored on the gateway, user will be automatically logged in using the login information (username and password pair) was used when the cookie was first generated. Otherwise, the user will be prompted to log in, and in the case authentication is successful, old cookie will be removed from the local HotSpot active cookie list and the new one with different random ID and expiration time will be added to the list and sent to the web browser. It is also possible to erase cookie on user manual logoff (not in the default server pages). This method may only be used together with HTTP PAP, HTTP CHAP or HTTPS methods as there would be nothing to generate cookies in the first place otherwise.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **MAC address** - try to authenticate clients as soon as they appear in the hosts list (i.e., as soon as they have sent any packet to the HotSpot server), using client's MAC address as username

There are currently 5 different authentication methods. You can use one or more of them simultaneously:

HotSpot can authenticate users consulting the local user database or a RADIUS server (local database is consulted first, then - a RADIUS server). In case of HTTP cookie authentication via RADIUS server, the router will send the same information to the server as was used when the cookie was first generated. If authentication is done locally, profile corresponding to that user is used, otherwise (in case RADIUS reply did not contain the group for that user) the default profile is used to set default values for parameters, which are not set in RADIUS access-accept message. For more information on how the interaction with a RADIUS server works, see the respective manual section.

The HTTP PAP method also makes it possible to authenticate by requesting the page `/login?username=username&password=password` . In case you want to log in using telnet connection, the exact HTTP request would look like that: **GET /login?username=username&password=password HTTP/1.0** (note that the request is case-sensitive)

Authorization

After authentication, user gets access to the Internet, and receives some limitations (which are user profile specific). HotSpot may also perform a one-to-one NAT for the client, so that a particular user would always receive the same IP address regardless of what PC is he/she working at.

The system will automatically detect and redirect requests to a proxy server a client is using (if any; it may be set in his/her settings to use an unknown to us proxy server) to the proxy server embedded in the router.

Advertisement

The same proxy used for unauthorized clients to provide Walled-Garden facility, may also be used for authorized users to show them advertisement popups. Transparent proxy for authorized users allows to monitor http requests of the clients and to take some action if required. It enables the possibility to open status page even if client is logged in by mac address, as well as to show advertisements time after time

When time has come to show an advertisement, the server redirects client's web browser to the status page. Only requests, which provide html content, are redirected (images and other content will not be affected). The status page displays the advertisement and next advertise-interval is used to schedule next advertisement. If status page is unable to display an advertisement for configured timeout starting from moment, when it is scheduled to be shown, client access is blocked within walled-garden (as unauthorized clients are). Client is unblocked when the scheduled page is finally shown. Note that if popup windows are blocked in the browser, the link on the status page may be used to open the advertisement manually.

While client is blocked, FTP and other services will not be allowed. Thus requiring client to open an advertisement for any Internet activity not especially allowed by the Walled-Garden.

Accounting

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

The HotSpot system implement accounting internally, you are not required to do anything special for it to work. The accounting information for each user may be sent to a RADIUS server.

Configuration menus

- **/ip hotspot** - HotSpot servers on particular interfaces (one server per interface). HotSpot server must be added in this menu in order for HotSpot system to work on an interface
- **/ip hotspot profile** - HotSpot server profiles. Settings, which affect login procedure for HotSpot clients are configured here. More than one HotSpot servers may use the same profile
- **/ip hotspot host** - dynamic list of active network hosts on all HotSpot interfaces. Here you can also find IP address bindings of the one-to-one NAT
- **/ip hotspot ip-binding** - rules for binding IP addresses to hosts on hotspot interfaces
- **/ip hotspot service-port** - address translation helpers for the one-to-one NAT
- **/ip hotspot walled-garden** - Walled Garden rules at HTTP level (DNS names, HTTP request substrings)
- **/ip hotspot walled-garden ip** - Walled Garden rules at IP level (IP addresses, IP protocols)
- **/ip hotspot user** - local HotSpot system users
- **/ip hotspot user profile** - local HotSpot system users profiles (user groups)
- **/ip hotspot active** - dynamic list of all authenticated HotSpot users
- **/ip hotspot cookie** - dynamic list of all valid HTTP cookies

Question&Answer-Based Setup

Command name: ***/ip hotspot setup***

Command Description

address pool of network (*name*) - IP address pool for the HotSpot network

dns name (*text*) - DNS domain name of the HotSpot gateway (will be statically configured on the local DNS proxy

dns servers (*IP address* | *IP address*) - DNS servers for HotSpot clients

hotspot interface (*name*) - interface to run HotSpot on

ip address of smtp server (*IP address*; default: **0.0.0.0**) - IP address of the SMTP server to redirect SMTP requests (TCP port 25) to

- **0.0.0.0** - no redirect

local address of network (*IP address*; default: **10.5.50.1/24**) - HotSpot gateway address for the interface

masquerade network (yes | no; default: **yes**) - whether to masquerade the HotSpot network

name of local hotspot user (*text*; default: **admin**) - username of one automatically created user

passphrase (*text*) - the passphrase of the certificate you are importing

password for the user (*text*) - password for the automatically created user

select certificate (*name* | *none* | *import-other-certificate*) - choose SSL certificate from the list of the imported certificates

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **none** - do not use SSL
- **import-other-certificate** - setup the certificates not imported yet, and ask this question again

Notes

Depending on current settings and answers to the previous questions, default values of following questions may be different. Some questions may disappear if they become redundant

Example

To configure HotSpot on ether1 interface (which is already configured with address of 192.0.2.1/25), and adding user admin with password rubbish:

```
[admin@MikroTik] > ip hotspot setup
hotspot interface: ether1
local address of network: 192.0.2.1/24
masquerade network: yes
address pool of network: 192.0.2.2-192.0.2.126
select certificate: none
ip address of smtp server: 0.0.0.0
dns servers: 192.0.2.254
dns name: hs.example.net
name of local hotspot user: admin
password for the user: rubbish
[admin@MikroTik] >
```

HotSpot Interface Setup

Home menu level: ***/ip hotspot***

Description

HotSpot system is put on individual interfaces. You can run completely different HotSpot configurations on different interfaces

Property Description

addresses-per-mac (*integer* | *unlimited*; default: **2**) - number of IP addresses allowed to be bind with any particular MAC address (it is a small chance to reduce denial of service attack based on taking over all free IP addresses)

- **unlimited** - number of IP addresses per one MAC address is not limited

address-pool (*name* | *none*; default: **none**) - IP address pool name for performing one-to-one NAT. You can choose not to use the one-to-one NAT

- **none** - do not perform one-to-one NAT for the clients of this HotSpot interface

HTTPS (*read-only: flag*) - whether the HTTPS service is actually running on the interface (i.e., it is set up in the server profile, and a valid certificate is imported in the router)

idle-timeout (*time* | *none*; default: **00:05:00**) - idle timeout (maximal period of inactivity) for unauthorized clients. It is used to detect, that client is not using outer networks (e.g. Internet), i.e., there is NO TRAFFIC coming from that client and going through the router. Reaching the timeout, user will be dropped of the host list, and the address used buy the user will be freed

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **none** - do not timeout idle users

ip-of-dns-name (*read-only: IP address*) - IP address of the HotSpot gateway's DNS name set in the HotSpot interface profile

keepalive-timeout (*time* | *none*; default: **none**) - keepalive timeout for unauthorized clients. Used to detect, that the computer of the client is alive and reachable. If check will fail during this period, user will be dropped of the host list, and the address used buy the user will be freed

- **none** - do not timeout unreachable users

profile (*name*; default: **default**) - default HotSpot profile for the interface

Command Description

reset-html (*name*) - overwrite the existing HotSpot servlet with the original HTML files. It is used if you have changed the servlet and it is not working after that

Notes

addresses-per-mac property works only if address pool is defined. Also note that in case you are authenticating users connected through a router, than all the IP addresses will seem to have come from one MAC address.

Example

To add HotSpot system to the **local** interface, allowing the system to do one-to-one NAT for each client (addresses from the **HS-real** address pool will be used for the NAT):

```
[admin@MikroTik] ip hotspot> add interface=local address-pool=HS-real
[admin@MikroTik] ip hotspot> print
Flags: X - disabled, I - invalid, S - HTTPS
 #   NAME                    INTERFACE    ADDRESS-POOL PROFILE IDLE-TIMEOUT
 0   hs-local                local        HS-real      default 00:05:00
[admin@MikroTik] ip hotspot>
```

HotSpot Server Profiles

Home menu level: ***/ip hotspot profile***

Property Description

dns-name (*text*) - DNS name of the HotSpot server. This is the DNS name used as the name of the HotSpot server (i.e., it appears as the location of the login page). This name will automatically be added as a static DNS entry in the DNS cache

hotspot-address (*IP address*; default: **0.0.0.0**) - IP address for HotSpot service

html-directory (*text*; default: **""**) - name of the directory (accessible with FTP), which stores the HTML servlet pages (when changed, the default pages are automatically copied into specified directory if it does not exist already)

http-cookie-lifetime (*time*; default: **3d**) - validity time of HTTP cookies

http-proxy (*IP address*; default: **0.0.0.0**) - the address of the proxy server the HotSpot service will use as a proxy server for all those requests intercepted by Universal Proxy system and not defined

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

in the /ip proxy direct list. If not specified, the address defined in parent-proxy parameter of /ip proxy. If that is absent too, the request will be resolved by the local proxy

login-by (*multiple choice: cookie* | *http-chap* | *http-pap* | *https* | *mac*; default: **cookie,http-chap**) - which authentication methods to use

- **cookie** - use HTTP cookies to authenticate, without asking user credentials. Other method will be used in case the client does not have cookie, or the stored username and password pair are not valid anymore since the last authentication. May only be used together with other HTTP authentication methods (HTTP-PAP, HTTP-CHAP or HTTPS), as in the other case there would be no way for the cookies to be generated in the first place
- **http-chap** - use CHAP challenge-response method with MD5 hashing algorithm for hashing passwords. This way it is possible to avoid sending clear-text passwords over an insecure network. This is the default authentication method
- **http-pap** - use plain-text authentication over the network. Please note that in case this method will be used, your user passwords will be exposed on the local networks, so it will be possible to intercept them
- **https** - use encrypted SSL tunnel to transfer user communications with the HotSpot server. Note that in order this to work, a valid certificate must be imported into the router (see a separate manual on certificate management)
- **mac** - try to use client's MAC address first as its username. If the matching MAC address exists in the local user database or on the RADIUS server, the client will be authenticated without asking to fill the login form

radius-accounting (yes | no; default: **yes**) - whether to send RADIUS server accounting information on each user once in a while (the "while" is defined in the radius-interim-update property)

radius-interim-update (*time* | *received*; default: **received**) - how often to sent cumulative accounting reports.

- **0s** - same as received
- **received** - use whatever value received from the RADIUS server

rate-limit (*text*; default: "") - Rate limitation in form of rx-rate[/tx-rate] [rx-burst-rate[/tx-burst-rate] [rx-burst-threshold[/tx-burst-threshold] [rx-burst-time[/tx-burst-time]]]]. All rates should be numbers with optional 'k' (1,000s) or 'M' (1,000,000s). If tx-rate is not specified, rx-rate is as tx-rate too. Same goes for tx-burst-rate and tx-burst-threshold and tx-burst-time. If both rx-burst-threshold and tx-burst-threshold are not specified (but burst-rate is specified), rx-rate and tx-rate is used as burst thresholds. If both rx-burst-time and tx-burst-time are not specified, 1s is used as default

smtp-server (*IP address*; default: **0.0.0.0**) - default SMTP server to be used to redirect unconditionally all user SMTP requests to

split-user-domain (yes | no; default: **no**) - whether to split username from domain name when the username is given in "user@domain" or in "domain\user" format

ssl-certificate (*name* | *none*; default: **none**) - name of the SSL certificate to use for HTTPS authentication. Not used for other authentication methods

use-radius (yes | no; default: **no**) - whether to use RADIUS to authenticate HotSpot users

Notes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

If **dns-name** property is not specified, **hotspot-address** is used instead. If **hotspot-address** is also absent, then both are to be detected automatically.

In order to use RADIUS authentication, the **/radius** menu must be set up accordingly.

Example

HotSpot User Profiles

Home menu level: ***/ip hotspot user profile***

Description

Article moved to: ***HotSpot AAA section***

HotSpot Users

Home menu level: ***/ip hotspot user***

Description

Article moved to: ***HotSpot AAA section***

HotSpot Active Users

Home menu level: ***/ip hotspot active***

Description

Article moved to: ***HotSpot AAA section***

HotSpot Cookies

Home menu level: ***/ip hotspot cookie***

Description

Cookies can be used for authentication in the Hotspot service

Property Description

domain (*read-only: text*) - domain name (if split from username)
expires-in (*read-only: time*) - how long the cookie is valid
mac-address (*read-only: MAC address*) - user's MAC address
user (*read-only: name*) - username

Notes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

There can be multiple cookies with the same MAC address. For example, there will be a separate cookie for each web browser on the same computer.

Cookies can expire - that's the way how it is supposed to be. Default validity time for cookies is **3** days (72 hours), but it can be changed for each individual HotSpot server profile, for example :

```
/ip hotspot profile set default http-cookie-lifetime=1d
```

Example

To get the list of valid cookies:

```
[admin@MikroTik] ip hotspot cookie> print
  # USER               DOMAIN           MAC-ADDRESS       EXPIRES-IN
  0 ex                                  01:23:45:67:89:AB 23h54m16s
[admin@MikroTik] ip hotspot cookie>
```

HTTP-level Walled Garden

Home menu level: ***/ip hotspot walled-garden***

Description

Walled garden is a system which allows unauthorized use of some resources, but requires authorization to access other resources. This is useful, for example, to give access to some general information about HotSpot service provider or billing options.

This menu only manages Walled Garden for HTTP and HTTPS protocols. Other protocols can also be included in Walled Garden, but that is configured elsewhere (in **/ip hotspot walled-garden ip**; see the next section of this manual for details)

Property Description

action (*allow* | *deny*; default: **allow**) - action to undertake if a packet matches the rule:

- **allow** - allow the access to the page without prior authorization
- **deny** - the authorization is required to access this page

dst-address (*IP address*) - IP address of the destination web server

dst-host (*wildcard*; default: "") - domain name of the destination web server (this is a wildcard)

dst-port (*integer*; default: "") - the TCP port a client has send the request to

method (*text*) - HTTP method of the request

path (*text*; default: "") - the path of the request (this is a wildcard)

server (*name*) - name of the HotSpot server this rule applied to

src-address (*IP address*) - IP address of the user sending the request

Notes

Wildcard properties (**dst-host** and **dst-path**) match a complete string (i.e., they will not match "example.com" if they are set to "example"). Available wildcards are '*' (match any number of any

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

characters) and '?' (match any one character). Regular expressions are also accepted here, but if the property should be treated as a regular expression, it should start with a colon (':').

Small hits in using regular expressions:

- \\ symbol sequence is used to enter \ character in console
- **\.** pattern means **.** only (in regular expressions single dot in pattern means any symbol)
- to show that no symbols are allowed before the given pattern, we use ^ symbol at the beginning of the pattern
- to specify that no symbols are allowed after the given pattern, we use **$** symbol at the end of the pattern

You can not use **path** property for HTTPS requests as router can not (and should not - that is what the HTTPS protocol was made for!) decrypt the request.

Example

To allow unauthorized requests to the **www.example.com** domain's **/paynow.html** page:

```
[admin@MikroTik] ip hotspot walled-garden> add path="/paynow.html" \
\... dst-host="www.example.com"
[admin@MikroTik] ip hotspot walled-garden> print
Flags: X - disabled, D - dynamic
 0   dst-host="www.example.com" path="/paynow.html" action=allow
[admin@MikroTik] ip hotspot walled-garden>
```

IP-level Walled Garden

Home menu level: ***/ip hotspot walled-garden ip***

Description

This menu is manages Walled Garden for generic IP requests. See the previous section for managing HTTP and HTTPS protocol specific properties (like the actual DNS name, HTTP method and path used in requests).

Property Description

action (*allow* | *deny*; default: **allow**) - action to undertake if a packet matches the rule:

- **allow** - allow the access to the page without prior authorization
- **deny** - the authorization is required to access this page

dst-address (*IP address*) - IP address of the destination web server

dst-host (*text*; default: "") - domain name of the destination web server (this is not a regular expression or a wildcard of any kind). The DNS name specified is resolved to a list of IP addresses when the rule is added, and all those IP addresses are used

dst-port (*integer*; default: "") - the TCP or UDP port (protocol MUST be specified explicitly in the protocol property) a client has send the request to

protocol (*integer* | *ddp* | *egp* | *encap* | *ggp* | *gre* | *hmp* | *icmp* | *idpr-cmtp* | *igmp* | *ipencap* | *ipip* |

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

ipsec-ah | *ipsec-esp* | *iso-tp4* | *ospf* | *pup* | *rdp* | *rspf* | *st* | *tcp* | *udp* | *vmtp* | *xns-idp* | *xtp*) - IP protocol name

server (*name*) - name of the HotSpot server this rule applied to

src-address (*IP address*) - IP address of the user sending the request

Example

One-to-one NAT static address bindings

Home menu level: ***/ip hotspot ip-binding***

Description

You can setup NAT translations statically based on either the original IP address (or IP network), or the original MAC address. You can also allow some addresses to bypass HotSpot authentication (i.e., they will be able work without having to log in to the network first) and completely block some addresses.

Property Description

address (*IP address* | *netmask*; default: "") - the original IP address or network of the client

mac-address (*MAC address*; default: "") - the source MAC address of the client

server (*name* | *all*; default: **all**) - the name of the server the client is connecting to

to-address (*IP address*; default: "") - IP address to translate the original client address to. If address property is given as network, this is the starting address for the translation (i.e., the first address is translated to to-address, address + 1 to to-address + 1, and so on)

type (*regular* | *bypassed* | *blocked*) - type of the static binding entry

- **regular** - perform a one-to-one NAT translation according to the values set in this entry
- **bypassed** - perform the translation, but exclude the client from having to log in to the HotSpot system
- **blocked** - the translation will not be preformed, and all packets from the host will be dropped

Notes

This is an ordered list, so you can put more specific entries on the top of the list for them to override the more common that appear lower.

Active Host List

Home menu level: ***/ip hotspot host***

Description

This menu shows all active network hosts that are connected to the HotSpot gateway. This list includes all one-to-one NAT translations

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Property Description

address (*read-only: IP address*) - the original IP address of the client

authorized (*read-only: flag*) - whether the client is successfully authenticated by the HotSpot system

blocked (*read-only: flag*) - true, if access is blocked within walled-garden because of expired advertisement timeout

bridge-port (*read-only: name*) - the actual physical interface, which the host is connected to. This is used when HotSpot service is put on a bridge interface to determine the host's actual port within the bridge.

bypass-hotspot (*read-only: flag*) - whether the client does not need to be authorized by the HotSpot system

bytes-in (*read-only: integer*) - how many bytes did the router receive from the client

bytes-out (*read-only: integer*) - how many bytes did the router send to the client

host-dead-time (*read-only: time*) - how long has the router not received any packets (including ARP replies, keepalive replies and user traffic) from this host

idle-time (*read-only: time*) - the amount of time has the user been idle

idle-timeout (*read-only: time*) - the exact value of idle-timeout that applies to this user. This property shows how long should the user stay idle for it to be logged off automatically

keepalive-timeout (*read-only: time*) - the exact value of keepalive-timeout that applies to this user. This property shows how long should the user's computer stay out of reach for it to be logged off automatically

mac-address (*read-only: MAC address*) - the actual MAC address of the user

packets-in (*read-only: integer*) - how many packets did the router receive from the client

packets-out (*read-only: integer*) - how many packets did the router send to the client

server (*read-only: name*) - name of the server, which the host is connected to

static (*read-only: flag*) - whether this translation has been taken from the static IP binding list

to-address (*read-only: IP address*) - what address is the original IP address of the host translated to

uptime (*read-only: time*) - current session time of the user (i.e., how long has the user been in the active host list)

Command Description

make-binding - copy a dynamic entry from this list to the static IP bindings list (*name*) - item number (*text*) - custom comment to the static entry to be created (*regular* | *bypassed* | *blocked*) - the type of the static entry

Service Port

Home menu level: ***/ip hotspot service-port***

Description

Just like for classic NAT, the HotSpot embedded one-to-one NAT 'breaks' some protocols that are

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

incompatible with address translation. To leave these protocols consistent, helper modules must be used. For the one-to-one NAT the only such a module is for FTP protocol.

Property Description

name (*read-only: name*) - protocol name

ports (*read-only: integer*) - list of the ports on which the protocol is working

Example

To set the FTP protocol uses both 20 and 21 TCP port:

```
 [admin@MikroTik] ip hotspot service-port> print
 Flags: X - disabled
   #   NAME                                                                    PORTS
   0   ftp                                                                     21
 [admin@MikroTik] ip hotspot service-port> set ftp ports=20,21
 [admin@MikroTik] ip hotspot service-port> print
 Flags: X - disabled
   #   NAME                                                                    PORTS
   0   ftp                                                                     20
                                                                               21
 [admin@MikroTik] ip hotspot service-port>
```

Customizing HotSpot: Firewall Section

Description

Apart from the obvious dynamic entries in the **/ip hotspot** submenu itself (like hosts and active users), some additional rules are added in the firewall tables when activating a HotSpot service. Unlike RouterOS version 2.8, there are relatively few firewall rules added in the firewall as the main job is made by the one-to-one NAT algorithm.

NAT rules

From **/ip firewall nat print dynamic** command, you can get something like this (comments follow after each of the rules):

Putting all HotSpot-related tasks for packets from all HotSpot clients into a separate chain

Redirect all DNS requests to the HotSpot service. The 64872 port provides DNS service for all HotSpot users. If you want HotSpot server to listen also to another port, add rules here the same way, changing **dst-port** property

Redirect all HTTP login requests to the HTTP login servlet. The 64873 is HotSpot HTTP servlet port.

Redirect all HTTPS login requests to the HTTPS login servlet. The 64875 is HotSpot HTTPS servlet port.

All other packets except DNS and login requests from unauthorized clients should pass through the **hs-unauth** chain

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

And packets from the authorized clients - through the **hs-auth** chain

First in the **hs-unauth** chain is put everything that affects TCP protocol in the **/ip hotspot walled-garden ip** submenu (i.e., everything where either protocol is not set, or set to TCP). Here we are excluding www.mikrotik.com from being redirected to the login page.

All other HTTP requests are redirected to the Walled Garden proxy server which listens the 64874 port. If there is an **allow** entry in the **/ip hotspot walled-garden** menu for an HTTP request, it is being forwarded to the destination. Otherwise, the request will be automatically redirected to the HotSpot login servlet (port 64873).

HotSpot by default assumes that only these ports may be used for HTTP proxy requests. These two entries are used to "catch" client requests to unknown proxies. I.e., to make it possible for the clients with unknown proxy settings to work with the HotSpot system. This feature is called "Universal Proxy". If it is detected that a client is using some proxy server, the system will automatically mark that packets with the **http** hotspot mark to work around the unknown proxy problem, as we will see later on. Note that the port used (64874) is the same as for HTTP requests in the rule #8 (so both HTTP and HTTP proxy requests are processed by the same code).

HTTPS proxy is listening on the 64875 port

Redirect for SMTP protocol may also be defined in the HotSpot configuration. In case it is, a redirect rule will be put in the **hs-smtp** chain. This is done so that users with unknown SMTP configuration would be able to send their mail through the service provider's (your) SMTP server instead of going to [possibly unavailable outside their network of origin] the SMTP server users have configured in their computers.

Providing HTTP proxy service for authorized users. Authenticated user requests may need to be subject to the transparent proxying (the "Universal Proxy" technique and for the advertisement feature). This **http** mark is put automatically on the HTTP proxy requests to the servers detected by the HotSpot HTTP proxy (the one that is listening on the 64874 port) to be HTTP proxy requests to unknown proxy servers. This is done so that users that have some proxy settings would use the HotSpot gateway instead of the [possibly unavailable outside their network of origin] proxy server users have configured in their computers. The mark is as well put on any HTTP requests done form the users whoose profile is configured to transparently proxy their requests.

Providing SMTP proxy for authorized users (the same as in rule #12)

Packet filter rules

From **/ip firewall filter print dynamic** command, you can get something like this (comments follow after each of the rules):

Any packet that traverse the router from unauthorized client will be sent to the **hs-unauth** chain. The **hs-unauth** implements the IP-based Walled Garden filter.

Everything that comes to clients through the router, gets redirected to another chain, called **hs-unauth-to**. This chain should reject unauthorized requests to the clients

Everything that comes from clients to the router itself, gets to another chain, called **hs-input**.

Allow client access to the local authentication and proxy services (as described earlier)

All other traffic from unauthorized clients to the router itself will be treated the same way as the

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

traffic traversing the routers

Unlike NAT table where only TCP-protocol related Walled Garden entries were added, in the packet filter **hs-unauth** chain is added everything you have set in the **/ip hotspot walled-garden ip** menu. That is why although you have seen only one entry in the NAT table, there are two rules here.

Everything else that has not been while-listed by the Walled Garden will be rejected. Note usage of TCP Reset for rejecting TCP connections.

Reject all packets to the clients with ICMP reject message

Customizing HotSpot: HTTP Servlet Pages

Description

You can create a completely different set of servlet pages for each HotSpot server you have, specifying the directory it will be stored in **html-directory** property of a HotSpot server profile (**/ip hotspot profile**). The default servlet pages are copied in the directory of your choice right after you create the profile. This directory can be accessed by connecting to the router with an FTP client. You can modify the pages as you like using the information from this section of the manual.

Available Servlet Pages

- **redirect.html** - redirects user to another url (for example, to login page)
- **login.html** - login page shown to a user to ask for username and password
- **md5.js** - javascript for md5 password encryption. Used together with http-chap login method
- **alogin.html** - page shown after client has logged in. It pops-up status page and redirects browser to originally requested page (before he/she was redirected to the HotSpot login page)
- **status.html** - status page, shows statistics for the client
- **logout.html** - logout page, shown after user is logged out. Shows final statistics about the finished session
- **error.html** - error page, shown on fatal errors only

Main HTML servlet pages, which are shown to user:

- **rlogin.html** - page, which redirects client from some other URL to the login page, if authorization of the client is required to access that URL
- **rstatus.html** - similarly to rlogin.html, only in case if the client is already logged in and the original URL is not known
- **flogin.html** - shown instead of login.html, if some error has happened (invalid username or password, for example)
- **fstatus.html** - shown instead of redirect, if status page is requested, but client is not logged in
- **flogout.html** - shown instead of redirect, if logout page is requested, but client is not logged in

Some other pages are available as well, if more control is needed:

Serving Servlet Pages

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

The HotSpot servlet recognizes 5 different request types:

1. request for a remote host
   - if user is logged in, the requested page is served
   - if user is not logged in, but the destination host is allowed by walled garden, then the request is also served
   - if user is not logged in, and the destination host is disallowed by walled garden, **rlogin.html** is displayed; if **rlogin.html** is not found, **redirect.html** is used to redirect to the login page
2. request for "/" on the HotSpot host
   - if user is logged in, **rstatus.html** is displayed; if **rstatus.html** is not found, **redirect.html** is used to redirect to the status page
   - if user is not logged in, **rlogin.html** is displayed; if **rlogin.html** is not found, **redirect.html** is used to redirect to the login page
3. request for "/login" page
   - if user has successfully logged in (or is already logged in), **alogin.html** is displayed; if **alogin.html** is not found, **redirect.html** is used to redirect to the originally requested page or the status page (in case, original destination page was not given)
   - if user is not logged in (username was not supplied, no error message appeared), **login.html** is showed
   - if login procedure has failed (error message is supplied), **flogin.html** is displayed; if **flogin.html** is not found, **login.html** is used
   - in case of fatal errors, **error.html** is showed
4. request for "/status" page
   - if user is logged in, **status.html** is displayed
   - if user is not logged in, **fstatus.html** is displayed; if **fstatus.html** is not found, **redirect.html** is used to redirect to the login page
5. request for '/logout' page
   - if user is logged in, **logout.html** is displayed
   - if user is not logged in, **flogout.html** is displayed; if **flogout.html** is not found, **redirect.html** is used to redirect to the login page

Note that if it is not possible to meet a request using the pages stored on the router's FTP server, Error 404 is displayed

There are many possibilities to customize what the HotSpot authentication pages look like:

- The pages are easily modifiable. They are stored on the router's FTP server in the directory you choose for the respective HotSpot server profile.
- By changing the variables, which client sends to the HotSpot servlet, it is possible to reduce keyword count to one (username or password; for example, the client's MAC address may be used as the other value) or even to zero (License Agreement; some predefined values general

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

for all users or client's MAC address may be used as username and password)

- Registration may occur on a different server (for example, on a server that is able to charge Credit Cards). Client's MAC address may be passed to it, so that this information need not be written in manually. After the registration, the server may change RADIUS database enabling client to log in for some amount of time.

To insert variable in some place in HTML file, the $(var_name) syntax is used, where the "var_name" is the name of the variable (without quotes). This construction may be used in any HotSpot HTML file accessed as '/', '/login', '/status' or '/logout', as well as any text or HTML file stored on the HotSpot server. For example, to show a link to the login page, following construction can be used:

Variables

All of the Servlet HTML pages use variables to show user specific values. Variable names appear only in the HTML source of the servlet pages - they are automatically replaced with the respective values by the HotSpot Servlet. For each variable there is an example of its possible value included in brackets. All the described variables are valid in all servlet pages, but some of them just might be empty at the time they are accesses (for example, there is no uptime before a user has logged in).

- Common server variables:
  - **hostname** - DNS name or IP address (if DNS name is not given) of the HotSpot Servlet ("hotspot.example.net")
  - **identity** - RouterOS identity name ("MikroTik")
  - **plain-passwd** - a "yes/no" representation of whether HTTP-PAP login method is allowed ("no")
  - **server-address** - HotSpot server address ("10.5.50.1:80")
  - **ssl-login** - a "yes/no" representation of whether HTTPS method was used to access that servlet page ("no")
  - **link-login** - link to login page including original URL requested ("http://10.5.50.1/login?dst=http://www.example.com/")
  - **link-login-plain** - link to login page, not including original URL requested ("http://10.5.50.1/login")
  - **link-logout** - link to logout page ("http://10.5.50.1/logout")
  - **link-status** - link to status page ("http://10.5.50.1/status")
  - **link-orig** - original URL requested ("http://www.example.com/")

    Links:

  - **domain** - domain name of the user ("mt.lv")
  - **ip** - IP address of the client ("10.5.50.2")
  - **logged-in** - "yes" if the user is logged in, otherwise - "no" ("yes")
  - **mac** - MAC address of the user ("01:23:45:67:89:AB")
  - **username** - the name of the user ("John")

    General client information

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **idle-timeout** - idle timeout ("20m" or "---" if none)
- **idle-timeout-secs** - idle timeout in seconds ("88" or "" if there is such timeout)
- **limit-bytes-in** - byte limit for send ("1000000" or "---" if there is no limit)
- **limit-bytes-out** - byte limit for receive ("1000000" or "---" if there is no limit)
- **refresh-timeout** - status page refresh timeout ("1m30s")
- **refresh-timeout-secs** - status page refresh timeout in seconds ("90s")
- **session-timeout** - session time left for the user ("5h" or "---" if none)
- **session-timeout-secs** - session time left for the user, in seconds ("3475" or "" if there is such timeout)
- **uptime** - current session uptime ("10h2m33s")
- **uptime-secs** - current session uptime in seconds ("125")

User status information:

- **bytes-in** - number of bytes received from the user ("15423")
- **bytes-in-nice** - user-friendly form of number of bytes received from the user ("15423")
- **bytes-out** - number of bytes sent to the user ("11352")
- **bytes-out-nice** - user-friendly form of number of bytes sent to the user ("11352")
- **packets-in** - number of packets received from the user ("251")
- **packets-out** - number of packets sent to the user ("211")
- **remain-bytes-in** - remaining bytes until limit-bytes-in will be reached ("337465" or "---" if there is no limit)
- **remain-bytes-out** - remaining bytes until limit-bytes-out will be reached ("124455" or "---" if there is no limit)

Traffic counters, which are available only in status page:

- **session-id** - value of 'session-id' parameter in the last request
- **var** - value of 'var' parameter in the last request
- **error** - error message, if something failed ("invalid username or password")
- **error-orig** - original error message (without translations retrieved from errors.txt), if something failed ("invalid username or password")
- **chap-id** - value of chap ID ("\371")
- **chap-challenge** - value of chap challenge ("\357\015\330\013\021\234\145\245\303\253\142\246\133\175\375\316")
- **popup** - whether to pop-up checkbox ("true" or "false")
- **advert-pending** - whether an advertisement is pending to be displayed ("yes" or "no")

Miscellaneous variables

Working with variables

`$(if <var_name>)` statements can be used in theses pages. Following content will be included, if value of <var_name> will not be an empty string. It is an equivalent to `$(if <var_name> != "")` It

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

is possible to compare on equivalence as well: `$(if <var_name> == <value>)` These statements have effect until `$(elif <var_name>)`, `$(else)` or `$(endif)`. In general case it looks like this:

Only one of those expressions will be shown. Which one - depends on values of those variables for each client.

Customizing Error Messages

All error messages are stored in the **errors.txt** file within the respective HotSpot servlet directory. You can change and translate all these messages to your native language. To do so, edit the **errors.txt** file. You can also use variables in the messages. All instructions are given in that file.

Multiple Versions of HotSpot Pages

Multiple hotspot page sets for the same hotspot server are supported. They can be chosen by user (to select language) or automatically by JavaScript (to select PDA/regular version of HTML pages).

To utilize this feature, create subdirectories in HotSpot HTML directory, and place those HTML files, which are different, in that subdirectory. For example, to translate everything in Latvian, subdirectory "lv" can be created with login.html, logout.html, status.html, alogin.html, radvert.html and errors.txt files, which are translated into Latvian. If the requested HTML page can not be found in the requested subdirectory, the corresponding HTML file from the main directory will be used. Then main login.html file would contain link to "/lv/login?dst=$(link-orig-esc)", which then displays Latvian version of login page: `<a href="/lv/login?dst=$(link-orig-esc)">Latviski</a>`. And Latvian version would contain link to English version: `<a href="/login?dst=$(link-orig-esc)">English</a>`

Another way of referencing directories is to specify 'target' variable:

After preferred directory has been selected (for example, "lv"), all links to local HotSpot pages will contain that path (for example, `$(link-status) = "http://hotspot.mt.lv/lv/status"`). So, if all hotspot pages reference links using "$(link-xxx)" variables, then no more changes are to be made - each client will stay within the selected directory all the time.

Notes

If you want to use HTTP-CHAP authentication method it is supposed that you include the **doLogin()** function (which references to the **md5.js** which must be already loaded) before the **Submit** action of the login form. Otherwise, CHAP login will fail.

The resulting password to be sent to the HotSpot gateway in case of HTTP-CHAP method, is formed MD5-hashing the concatenation of the following: chap-id, the password of the user and chap-challenge (in the given order)

In case if variables are to be used in link directly, then they must be escaped accordingly. For example, in login page, link will not work as intended, if username will be "123&456=1 2". In this case instead of $(user), its escaped version must be used: $(user-esc): link. Now the same username will be converted to "123%26456%3D1+2", which is the valid representation of "123&456=1 2" in URL. This trick may be used with any variables, not only with $(username).

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

There is a boolean parameter "erase-cookie" to the logout page, which may be either "on" or "true" to delete user cookie on logout (so that the user would not be automatically logged on when he/she opens a browser next time.

Example

With basic HTML language knowledge and the examples below it should be easy to implement the ideas described above.

- To provide predefined value as username, in login.html change:

```
<type="text" value="$(username)>
```

  to this line:

```
<input type="hidden" name="user" value="hsuser">
```

  (where **hsuser** is the username you are providing)

- To provide predefined value as password, in login.html change:

```
<input type="password">
```

  to this line:

```
<input type="hidden" name="password" value="hspass">
```

  (where **hspass** is the password you are providing)

- To send client's MAC address to a registration server in form of:

```
https://www.server.serv/register.html?mac=XX:XX:XX:XX:XX:XX
```

  change the Login button link in login.html to:

```
https://www.server.serv/register.html?mac=$(mac)
```

  (you should correct the link to point to your server)

- To show a banner after user login, in alogin.html after

```
$(if popup == 'true')
```

  add the following line:

```
open('http://your.web.server/your-banner-page.html', 'my-banner-name','');
```

  (you should correct the link to point to the page you want to show)

- To choose different page shown after login, in login.html change:

```
<input type="hidden" name="dst" value="$(link-orig)">
```

  to this line:

```
<input type="hidden" name="dst" value="http://your.web.server">
```

  (you should correct the link to point to your server)

- To erase the cookie on logoff, in the page containing link to the logout (for example, in status.html) change:

```
open('$(link-logout)', 'hotspot_logout', ...
```

  to this:

```
open('$(link-logout)?erase-cookie=on', 'hotspot_logout', ...
```

  or alternatively add this line:

```
<input type="hidden" name="erase-cookie" value="on">
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

before this one:

```
<input type="submit" value="log off">
```

An another example is making HotSpot to authenticate on a remote server (which may, for example, perform creditcard charging):

- Allow direct access to the external server in walled-garden (either HTTP-based, or IP-based)
- Modify login page of the HotSpot servlet to redirect to the external authentication server. The external server should modify RADIUS database as needed
  Here is an example of such a login page to put on the HotSpot router (it is redirecting to https://auth.example.com/login.php, replace with the actual address of an external authentication server):
  ```
  <html> <title>...</title> <body> <form name="redirect"
  action="https://auth.example.com/login.php" method="post"> <input type="hidden"
  name="mac" value="$(mac)"> <input type="hidden" name="ip" value="$(ip)"> <input
  type="hidden" name="user" value="$(username)"> <input type="hidden"
  name="link-login" value="$(link-login)"> <input type="hidden" name="link-orig"
  value="$(link-orig)"> <input type="hidden" name="error" value="$(error)"> </form>
  <script language="JavaScript"> <!-- document.redirect.submit(); //--> </script>
  </body> </html>
  ```
- The external server can log in a HotSpot client by redirecting it back to the original HotSpot servlet login page, specifying the correct username and password
  Here is an example of such a page (it is redirecting to https://hotspot.example.com/login, replace with the actual address of a HotSpot router; also, it is displaying www.mikrotik.com after successful login, replace with what needed):
  ```
  <html> <title>Hotspot login page</title> <body> <form name="login"
  action="https://hotspot.example.com/login" method="post"> <input type="text"
  name="username" value="demo"> <input type="password" name="password" value="none">
  <input type="hidden" name="domain" value=""> <input type="hidden" name="dst"
  value="http://www.mikrotik.com/"> <input type="submit" name="login" value="log in">
  </form> </body> </html>
  ```
- Hotspot will ask RADIUS server whether to allow the login or not. If not allowed, alogin.html page will be displayed (it can be modified to do anything!). If not allowed, flogin.html (or login.html) page will be displayed, which will redirect client back to the external authentication server.
- Note: as shown in these examples, HTTPS protocol and POST method can be used to secure communications.

Possible Error Messages

Description

There are two kinds of errors: fatal non-fatal. Fatal errors are shown on a separate HTML page called error.html. Non-fatal errors are basically indicating incorrect user actions and are shown on the login form.

General non-fatal errors:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **You are not logged in** - trying to access the status page or log off while not logged in. Solution: log in
- **already authorizing, retry later** - authorization in progress. Client already has issued an authorization request which is not yet complete. Solution: wait for the current request to be completed, and then try again
- **chap-missing = web browser did not send challenge response (try again, enable JavaScript)** - trying to log in with HTTP-CHAP method using MD5 hash, but HotSpot server does not know the challenge used for the hash. This may happen if you use BACK buttons in browser; if JavaScript is not enabled in web browser; if login.html page is not valid; or if challenge value has expired on server (more than 1h of inactivity). Solution: instructing browser to reload (refresh) the login page usually helps if JavaScript is enabled and login.html page is valid
- **invalid username ($(username)): this MAC address is not yours** - trying to log in using a MAC address username different from the actual user's MAC address. Solution: no - users with usernames that look like a MAC address (eg., 12:34:56:78:9a:bc) may only log in from the MAC address specified as their user name
- **session limit reached ($(error-orig))** - depending on licence number of active hotspot clients is limited to some number. The error is displayed when this limit is reached. Solution: try to log in later when there will be less concurrent user sessions, or buy an another license that allows more simultaneous sessions
- **hotspot service is shutting down** - RouterOS is currently being restarted or shut down. Solution: wait until the service will be available again

General fatal errors:

- **internal error ($(error-orig))** - this should never happen. If it will, error page will be shown displaying this error message (error-orig will describe what has happened). Solution: correct the error reported
- **configuration error ($(error-orig))** - the HotSpot server is not configured properly (error-orig will describe what has happened). Solution: correct the error reported
- **cannot assign ip address - no more free addresses from pool** - unable to get an IP address from an IP pool as there is no more free IP addresses in that pool. Solution: make sure there is a sufficient amount of free IP addresses in IP pool

Local HotSpot user database non-fatal errors:

- **invalid username or password** - self-explanatory
- **user $(username) is not allowed to log in from this MAC address** - trying to log in from a MAC address different from specified in user database. Solution: log in from the correct MAC address or take out the limitation
- **user $(username) has reached uptime limit** - self-explanatory
- **user $(username) has reached traffic limit** - either limit-bytes-in or limit-bytes-out limit is reached
- **no more sessions are allowed for user $(username)** - the shared-users limit for the user's profile is reached. Solution: wait until someone with this username logs out, use different login name or extend the shared-users limit

RADIUS client non-fatal errors:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **invalid username or password** - RADIUS server has rejected the username and password sent to it without specifying a reason. Cause: either wrong username and/or password, or other error. Solution: should be clarified in RADIUS server's log files
- **<error_message_sent_by_radius_server>** - this may be any message (any text string) sent back by RADIUS server. Consult with your RADIUS server's documentation for further information

RADIUS client fatal errors:

- **RADIUS server is not responding** - user is being authenticated by RADIUS server, but no response is received from it. Solution: check whether the RADIUS server is running and is reachable from the HotSpot router

HotSpot How-to's

Description

This section will focus on some simple examples of how to use your HotSpot system, as well as give some useful ideas.

Setting up https authorization

At first certificate must be present with decrypted private key:

Then we can use that certificate for hotspot:

After that we can see, that HTTPS is running on hotspot interface:

Bypass hotspot for some devices in hotspot network

All IP binding entries with **type** property set to **bypassed**, will not be asked to authorize - it means that they will have login-free access:

If all fields has been filled in the ip-binding table and **type** has been set to **bypassed**, then the IP address of this entry will be accessible from public interfaces immediately:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

HTTP Proxy

Document revision 1.1 (Wed Jul 06 09:55:38 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Quick Setup Guide
  Specifications
  Related Documents
  Description
Setup
  Property Description
  Notes
  Example
Access List
  Description
  Property Description
  Notes
Direct Access List
  Description
  Property Description
  Notes
HTTP Methods
  Description

General Information

Summary

The MikroTik RouterOS implements the following proxy server features:

- Regular HTTP proxy
- Transparent proxy. Can be transparent and regular at the same time
- Access list by source, destination, URL and requested method
- Direct Access List (specifies which resources should be accessed directly, and which - through another proxy server)
- Logging facility

Quick Setup Guide

To enable HTTP proxy, do the following:

```
[admin@MikroTik] ip proxy> set enabled=yes
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] ip proxy> print
                    enabled: yes
                       port: 8080
              parent-proxy: 0.0.0.0:0
  maximal-client-connecions: 1000
  maximal-server-connectons: 1000
[admin@MikroTik] ip proxy>
```

Remember to secure your proxy by preventing unauthorized access to it. Also you need to setup destination NAT in order to utilize transparent proxying facility:

```
[admin@MikroTik] ip firewall nat> add chain=dstnat protocol=tcp dst-port=80
action=redirect to-ports=8080
[admin@MikroTik] ip firewall nat> print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=dstnat protocol=tcp dst-port=80 action=redirect to-ports=8080
[admin@MikroTik] ip firewall nat>
```

Specifications

Packages required: ***system***
License required: ***level3***
Home menu level: ***/ip proxy***
Standards and Technologies: ***HTTP/1.0***, ***HTTP/1.1***, ***FTP***

Related Documents

- ***Software Package Management***
- ***IP Addresses and ARP***
-

Description

This service performs proxying of HTTP and HTTP-proxy (for FTP, HTTP and HTTPS protocols) requests. It does not support caching transactions yet.

When setting up proxy service, make sure it serves only your clients, and is not misused as relay. Please read the security notice in the Access List Section!

Note that the proxy may be used as something like HTTP and FTP firewall (for example, denying access to mp3 files) or to redirect requests to external proxy (possibly, to a proxy with caching functions) transparently.

Setup

Home menu level: ***/ip proxy***

Property Description

enabled (yes | no; default: **no**) - whether the proxy server is enabled

maximal-client-connecions (*integer*; default: **1000**) - maximal number of connections accepted from clients (any further connections will be rejected)

maximal-server-connectons (*integer*; default: **1000**) - maximal number of connections made to

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

servers (any further connections from clients will be put on hold until some server connections will terminate)

parent-proxy (*IP address* | *port*; default: **0.0.0.0:0**) - IP address and port of another HTTP proxy to redirect all requests to (exceptions may be defined in the "direct access" list

- **0.0.0.0:0** - no parent proxy is used

port (*port*; default: **8080**) - TCP port the proxy server will be listening on. This is to be specified on all clients that want to use the server as HTTP proxy. Transparent (with zero configuration for clients) proxy setup can be made by redirecting HTTP requests to this port in IP firewall using destination NAT feature

Notes

The web proxy listens to all IP addresses that the router has in its IP address list.

Example

To enable the proxy on port 8000:

```
[admin@MikroTik] ip proxy> set enabled=yes port=8000
[admin@MikroTik] ip proxy> print
                   enabled: yes
                      port: 8000
              parent-proxy: 0.0.0.0:0
  maximal-client-connecions: 1000
  maximal-server-connectons: 1000
[admin@MikroTik] ip proxy>
```

Access List

Home menu level: ***/ip proxy access***

Description

Access list is implemented in the same way as MikroTik RouterOS firewall rules. Rules are processed from the top to the bottom. First matching rule specifies decision of what to do with this connection. There is a total of 6 classifiers that specify matching constraints. If none of these classifiers is specified, the particular rule will match every connection.

If connection is matched by a rule, **action** property of this rule specifies whether connection will be allowed or not. If the particular connection does not match any rule, it will be allowed.

Property Description

action (*allow* | *deny*; default: **allow**) - specifies whether to pass or deny matched packets

dst-address (*IP address* | *netmask*) - destination address of the IP packet

dst-host (*wildcard*) - IP address or DNS name used to make connection the target server (this is the string user wrote in his/her browser before specifying port and path to a particular web page)

dst-port (*port*) - a list or range of ports the packet is destined to

method (*any* | *connect* | *delete* | *get* | *head* | *options* | *post* | *put* | *trace*) - HTTP method used in the request (see HTTP Methods section in the end of this document)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

path (*wildcard*) - name of the requested page within the target server (i.e. the name of a particular wed page or document without the name of the server it resides on)

src-address (*IP address* | *netmask*) - source address of the IP packet

Notes

Wildcard properties (**dst-host** and **dst-path**) match a complete string (i.e., they will not match "example.com" if they are set to "example"). Available wildcards are '*' (match any number of any characters) and '?' (match any one character). Regular expressions are also accepted here, but if the property should be treated as a regular expression, it should start with a colon (':').

Small hits in using regular expressions:

- \\ symbol sequence is used to enter \ character in console
- **\.** pattern means **.** only (in regular expressions single dot in pattern means any symbol)
- to show that no symbols are allowed before the given pattern, we use ^ symbol at the beginning of the pattern
- to specify that no symbols are allowed after the given pattern, we use **$** symbol at the end of the pattern
- to enter **[** or **]** symbols, you should escape them with backslash \.

It is strongly recommended to deny all IP addresses except those behind the router as the proxy still may be used to access your internal-use-only (intranet) web servers. Also, consult examples in Firewall Manual on how to protect your router.

Direct Access List

Home menu level: ***/ip proxy direct***

Description

If **parent-proxy** property is specified, it is possible to tell proxy server whether to try to pass the request to the parent proxy or to resolve it connecting to the requested server directly. Direct Access List is managed just like Proxy Access List described in the previous chapter except the **action** argument.

Property Description

action (*allow* | *deny*; default: **allow**) - specifies the action to perform on matched packets

- **allow** - always resolve matched requests directly bypassing the parent router
- **deny** - resolve matched requests through the parent proxy. If no one is specified this has the same effect as allow

dst-address (*IP address* | *netmask*) - destination address of the IP packet

dst-host (*text*) - IP address or DNS name used to make connection the target server (this is the string user wrote in his/her browser before specifying port and path to a particular web page)

dst-port (*port*) - a list or range of ports the packet is destined to

method (*any* | *connect* | *delete* | *get* | *head* | *options* | *post* | *put* | *trace*) - HTTP method used in the

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

request (see HTTP Methods section in the end of this document)

path (*wildcard*) - name of the requested page within the target server (i.e. the name of a particular wed page or document without the name of the server it resides on)

src-address (*IP address* | *netmask*) - source address of the IP packet

Notes

Unlike the access list, the direct proxy access list has default action equal to **deny**. It takes place when no rules are specified or a particular request did not match any rule.

HTTP Methods

Description

OPTIONS

This method is a request of information about the communication options available on the chain between the client and the server identified by the **Request-URI**. The method allows the client to determine the options and (or) the requirements associated with a resource without initiating any resource retrieval

GET

This method retrieves whatever information identified by the **Request-URI**. If the **Request-URI** refers to a data processing process than the response to the **GET** method should contain data produced by the process, not the source code of the process procedure(-s), unless the source is the result of the process.

The **GET** method can become a *conditional* **GET** if the request message includes an **If-Modified-Since, If-Unmodified-Since, If-Match, If-None-Match**, or **If-Range** header field. The conditional **GET** method is used to reduce the network traffic specifying that the transfer of the entity should occur only under circumstances described by conditional header field(-s).

The **GET** method can become a *partial* **GET** if the request message includes a **Range** header field. The partial **GET** method intends to reduce unnecessary network usage by requesting only parts of entities without transferring data already held by client.

The response to a **GET** request is cacheable if and only if it meets the requirements for HTTP caching.

HEAD

This method shares all features of **GET** method except that the server must not return a message-body in the response. This retrieves the metainformation of the entity implied by the request which leads to a wide usage of it for testing hypertext links for validity, accessibility, and recent modification.

The response to a **HEAD** request may be cacheable in the way that the information contained in the

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

response may be used to update previously cached entity identified by that **Request-URI**.

POST

This method requests that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the **Request-URI**.

The actual action performed by the **POST** method is determined by the origin server and usually is **Request-URI** dependent.

Responses to **POST** method are not cacheable, unless the response includes appropriate **Cache-Control** or **Expires** header fields.

PUT

This method requests that the enclosed entity be stored under the supplied **Request-URI**. If another entity exists under specified **Request-URI**, the enclosed entity should be considered as updated (newer) version of that residing on the origin server. If the **Request-URI** is not pointing to an existing resource, the origin server should create a resource with that URI.

If the request passes through a cache and the **Request-URI** identifies one or more currently cached entities, those entries should be treated as stale. Responses to this method are not cacheable.

TRACE

This method invokes a remote, application-layer loop-back of the request message. The final recipient of the request should reflect the message received back to the client as the entity-body of a 200 (OK) response. The final recipient is either the origin server or the first proxy or gateway to receive a **Max-Forwards** value of **0** in the request. A **TRACE** request must not include an entity.

Responses to this method MUST NOT be cached.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

IP Pools

Document revision 0.0 (Thu Mar 04 20:47:26 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
  Notes
Setup
  Property Description
  Example
Used Addresses from Pool
  Description
  Property Description
  Example

General Information

Summary

IP pools are used to define range of IP addresses that is used for DHCP server and Point-to-Point servers

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip pool***
Standards and Technologies: ***none***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***IP Addresses and ARP***
- ***AAA***
- ***DHCP Client and Server***
- ***HotSpot Gateway***
- ***Universal Client Interface***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

IP pools simply group IP addresses for further usage. It is a single configuration point for all features that assign IP addresses to clients.

Notes

Whenever possible, the same ip address is given out to each client (OWNER/INFO pair).

Setup

Home menu level: ***/ip pool***

Property Description

name (*name*) - the name of the pool

next-pool (*name*) - when address is acquired from pool that has no free addresses, and next-pool property is set to another pool, then next IP address will be acquired from next-pool

ranges (*IP address*) - IP address list of non-overlapping IP address ranges in form of: from1-to1,from2-to2,...,fromN-toN. For example, 10.0.0.1-10.0.0.27,10.0.0.32-10.0.0.47

Example

To define a pool named **ip-pool** with the **10.0.0.1-10.0.0.125** address range excluding gateway's address **10.0.0.1** and server's address **10.0.0.100**, and the other pool **dhcp-pool**, with the **10.0.0.200-10.0.0.250** address range:

```
[admin@MikroTik] ip pool> add name=ip-pool ranges=10.0.0.2-10.0.0.99,10.0.0.101
10.0.0.126
[admin@MikroTik] ip pool> add name=dhcp-pool ranges=10.0.0.200-10.0.0.250
[admin@MikroTik] ip pool> print
  # NAME                                      RANGES
  0 ip-pool                                   10.0.0.2-10.0.0.99
                                              10.0.0.101-10.0.0.126
  1 dhcp-pool                                 10.0.0.200-10.0.0.250

[admin@MikroTik] ip pool>
```

Used Addresses from Pool

Home menu level: ***/ip pool used***

Description

Here you can see all used IP addresses from IP pools.

Property Description

pool (*read-only: name*) - name of the IP pool

address (*read-only: IP address*) - IP address that is assigned to client form the pool

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

owner (*read-only: MAC address*) - MAC address of the client

info (*read-only: name*) - name of the interface to which the client is connected to

Example

See used addresses from pool:

```
[admin@MikroTik] ip pool used> print
POOL  ADDRESS         OWNER                     INFO
local 192.168.0.100   00:0C:42:03:1F:60         test
local 192.168.0.99    00:0C:42:03:21:0F         test
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

SOCKS Proxy Server

Document revision 1.3 (Fri Apr 15 17:51:27 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
  Notes
  Additional Documents
SOCKS Configuration
  Description
  Property Description
  Example
Access List
  Description
  Property Description
Active Connections
  Description
  Property Description
  Example
  FTP service through SOCKS server

General Information

Summary

This manual discusses the SOCKS proxy server which is implemented in RouterOS. MikroTik RouterOS supports SOCKS version 4.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip socks***
Standards and Technologies: ***SOCKS version 4***
Hardware usage: ***Not significant***

Related Documents

- ***Web Proxy***
- ***NAT***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

SOCKS is a proxy server that allows TCP based application data to relay across the firewall, even if the firewall would block the packets. The SOCKS protocol is independent from application protocols, so it can be used for many services, e.g, WWW, FTP, TELNET, and others.

At first, an application client connects to the SOCKS proxy server, then the proxy server looks in its **access** list to see whether the client is permited to access the remote application server or not, if it is permitted, the proxy server relies the packet to the application server and creates a connection between the application server and client.

Notes

Remember to configure your application client to use SOCKS version 4.

You should secure the SOCKS proxy using its access list and/or firewall to disallow access from outisde. Failing to secure the proxy server may introduce security issues to your network, and may provide a way for spammers to send junk mail through the router.

Additional Documents

- ***Information about SOCKS***

SOCKS Configuration

Description

In this section you will learn how to enable the SOCKS proxy server and do its configuration.

Property Description

connection-idle-timeout (*time*; default: **2m**) - time after which idle connections are terminated

enabled (yes | no; default: **no**) - whether to enable or no the SOCKS proxy

max-connections (*integer*: 1..500; default: **200**) - maxumum number of simultaneous connections

port (*integer*: 1..65535; default: **1080**) - TCP port on which the SOCKS server listens for connections

Example

To enable SOCKS:

```
[admin@MikroTik] ip socks> set enabled=yes
[admin@MikroTik] ip socks> print
                    enabled: yes
                       port: 1080
    connection-idle-timeout: 2m
            max-connections: 200
[admin@MikroTik] ip socks>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Access List

Home menu level: ***/ip socks access***

Description

In the SOCKS access list you can add rules which will control access to SOCKS server. This list is similar to firewall lists.

Property Description

action (*allow* | *deny*; default: **allow**) - action to be performed for this rule

- **allow** - allow packets, matching this rule to be forwarded for further processing
- **deny** - deny access for packets, matching this rule

dst-address (*IP address* | *netmask* | *port*) - destination (server's) address

src-address (*IP address* | *netmask* | *port*) - source (client's) address for a packet

Active Connections

Home menu level: ***/ip socks connections***

Description

The Active Connection list shows all established TCP connections, which are maintained through the SOCKS proxy server.

Property Description

dst-address (*read-only: IP address*) - destination (application server) IP address

RX (*read-only: integer*) - bytes received

src-address (*read-only: IP address*) - source (application client) IP address

TX (*read-only: integer*) - bytes sent

Example

To see current TCP connections:

```
[admin@MikroTik] ip socks connections> print
 # SRC-ADDRESS               DST-ADDRESS               TX         RX
 0 192.168.0.2:3242          159.148.147.196:80        4847       2880
 1 192.168.0.2:3243          159.148.147.196:80        3408       2127
 2 192.168.0.2:3246          159.148.95.16:80          10172      25207
 3 192.168.0.2:3248          194.8.18.26:80            474        1629
 4 192.168.0.2:3249          159.148.95.16:80          6477       18695
 5 192.168.0.2:3250          159.148.95.16:80          4137       27568
 6 192.168.0.2:3251          159.148.95.16:80          1712       14296
 7 192.168.0.2:3258          80.91.34.241:80           314        208
 8 192.168.0.2:3259          80.91.34.241:80           934        524
 9 192.168.0.2:3260          80.91.34.241:80           930        524
10 192.168.0.2:3261          80.91.34.241:80           312        158
11 192.168.0.2:3262          80.91.34.241:80           312        158
[admin@MikroTik] ip socks connections>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

General Information

FTP service through SOCKS server

Let us consider that we have a network **192.168.0.0/24** which is masqueraded, using a router with a public IP **10.1.0.104/24** and a private IP **192.168.0.1/24**. Somewhere in the network is an FTP server with IP address **10.5.8.8**. We want to allow access to this FTP server for a client in our local network with IP address **192.168.0.2/24**.

We have already masqueraded our local network:

```
[admin@MikroTik] ip firewall nat> print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=srcnat src-address=192.168.0.0/24 action=masquerade
[admin@MikroTik] ip firewall nat>
```

And the access to public FTP servers is denied in firewall:

```
[admin@MikroTik] ip firewall filter> print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward src-address=192.168.0.0/24 dst-address=:21 action=drop
[admin@MikroTik] ip firewall filter>
```

We need to enable the SOCKS server:

```
[admin@MikroTik] ip socks> set enabled=yes
[admin@MikroTik] ip socks> print
                   enabled: yes
                      port: 1080
    connection-idle-timeout: 2m
           max-connections: 200
[admin@MikroTik] ip socks>
```

Add access to a client with an IP address **192.168.0.2/32** to SOCKS access list, allow data transfer from FTP server to client (allow destionation ports from 1024 to 65535 for any IP address), and drop everything else:

```
[admin@MikroTik] ip socks access> add src-address=192.168.0.2/32 dst-address=:21
action=allow
[admin@MikroTik] ip socks access> add dst-address=:1024-65535 action=allow
[admin@MikroTik] ip socks access> add action=deny
[admin@MikroTik] ip socks access> print
Flags: X - disabled
 0   src-address=192.168.0.2/32 dst-address=:21 action=allow
 1   dst-address=:1024-65535 action=allow
 2   action=deny
[admin@MikroTik] ip socks access>
```

That's all - the SOCKS server is configured. To see active connections and data transmitted and received:

```
[admin@MikroTik] ip socks connections> print
 # SRC-ADDRESS               DST-ADDRESS               TX         RX
 0 192.168.0.2:1238          10.5.8.8:21               1163       4625
 1 192.168.0.2:1258          10.5.8.8:3423             0          3231744
[admin@MikroTik] ip socks connections>
```

Note! In order to use SOCKS proxy server, you have to specify its IP address and port in your FTP

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

client. In this case IP address would be **192.168.0.1** (router's/SOCKS server's local IP) and port **1080**.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

UPnP

Document revision 2.2 (Tue Mar 08 19:21:08 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Description
  Additional Documents
Enabling Universal Plug-n-Play
  Property Description
  Example
UPnP Interfaces
  Property Description
  Notes
  Example

General Information

Summary

The MikroTik RouterOS supports Universal Plug and Play architecture for transparent peer-to-peer network connectivity of personal computers and network-enabled intelligent devices or appliances. UPnP builds enables these devices to automatically connect with one another and work together to make networking possible for more people.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip upnp***
Standards and Technologies: ***TCP/IP***, ***HTTP***, ***XML***, ***IGD***
Hardware usage: ***Not significant***

Description

UPnP enables data communication between any two devices under the command of any control device on the network. Universal Plug and Play is completely independent of any particular physical medium. It supports networking with automatic discovery without any initial configuration, whereby a device can dynamically join a network. DHCP and DNS servers are optional and will be used if available on the network. UPnP implements simple yet powerfull NAT traversal solution, that enables the client to get full peer-to-peer network support from behind the NAT.

There are two interface types for UPnP: internal (the one local clients are connected to) and external (the one the Internet is connected to). A router may only have one external interface with a 'public'

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

IP address on it, and as many internal IP addresses as needed, all with source-NATted 'internal' IP addresses.

The UPnP protocol is used for most of DirectX games as well as for various Windows Messenger features (remote asisstance, application sharing, file transfer, voice, video) from behind a firewall.

Additional Documents

Enabling Universal Plug-n-Play

Home menu level: ***/ip upnp***

Property Description

allow-disable-external-interface (yes | no; default: **yes**) - whether or not should the users be allowed to disable router's external interface. This functionality (for users to be able to turn the router's external interface off without any authentication procedure) is required by the standard, but as it is sometimes not expected or unwanted in UPnP deployments which the standard was not designed for (it was designed mostly for home users to establish their local networks), you can disable this behavior

enabled (yes | no; default: **no**) - whether UPnP feature is enabled

show-dummy-rule (yes | no; default: **yes**) - this is to enable a workaround for some broken implementations, which are handling the absense of UPnP rules inincorrectly (for example, popping up error messages). This option will instruct the server to install a dummy (meaningless) UPnP rule that can be observed by the clients, which refuse to work correctly otherwise

Example

To enable UPnP feature:

```
[admin@MikroTik] ip upnp> set enable=yes
[admin@MikroTik] ip upnp> print
                          enabled: yes
    allow-disable-external-interface: yes
                     show-dummy-rule: yes
[admin@MikroTik] ip upnp>
```

UPnP Interfaces

Home menu level: ***/ip upnp interfaces***

Property Description

interface (*name*) - interface name UPnP will be run on

type (*external* | *internal*) - interface type, one of the:

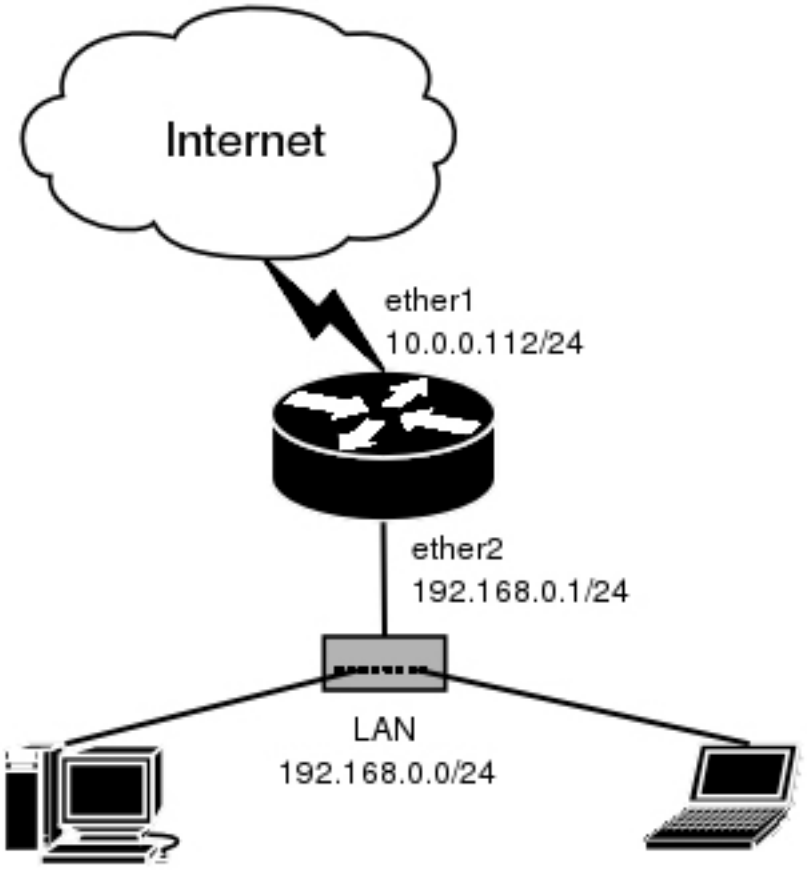
- **external** - the interface global IP address is assigned to
- **internal** - router's local interface

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Notes

It is highly recommended to upgrade DirectX runtime libraries to version ***DirectX 9.0a*** or higher and Windows Messenger to version***Windows Messenger 5.0*** or higher in order to get UPnP to work properly.

Example



We have masquerading already enabled on our router:

```
[admin@MikroTik] ip upnp interfaces> /ip firewall src-nat print
Flags: X - disabled, I - invalid, D - dynamic
  0   src-address=0.0.0.0/0:0-65535 dst-address=0.0.0.0/0:0-65535
      out-interface=ether1 protocol=all icmp-options=any:any flow=""
      connection="" content="" limit-count=0 limit-burst=0 limit-time=0s
      action=masquerade to-src-address=0.0.0.0 to-src-port=0-65535

[admin@MikroTik] ip upnp interfaces>
```

Now all we have to do is to add interfaces and enable UPnP:

```
[admin@MikroTik] ip upnp interfaces> add interface=ether1 type=external
[admin@MikroTik] ip upnp interfaces> add interface=ether2 type=internal
[admin@MikroTik] ip upnp interfaces> print
Flags: X - disabled
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  #   INTERFACE TYPE
  0 X ether1    external
  1 X ether2    internal

[admin@MikroTik] ip upnp interfaces> enable 0,1
[admin@MikroTik] ip upnp interfaces> .. set enabled=yes
[admin@MikroTik] ip upnp interfaces>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Web Proxy

Document revision 1.1 (Wed Jul 06 10:01:47 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Quick Setup Guide
  Specifications
  Related Documents
  Description
Setup
  Property Description
  Notes
  Example
Access List
  Description
  Property Description
  Notes
  Example
Direct Access List
  Description
  Property Description
  Notes
Cache Management
  Description
  Property Description
Complementary Tools
  Description
  Command Description
Transparent Mode
  Description
  Notes
  Example
HTTP Methods
  Description

General Information

Summary

The MikroTik RouterOS implements the following proxy server features:

- Regular HTTP proxy
- Transparent proxy. Can be transparent and regular at the same time

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- Access list by source, destination, URL and requested method
- Cache access list (specifies which objects to cache, and which not)
- Direct Access List (specifies which resources should be accessed directly, and which - through another proxy server)
- Logging facility

Quick Setup Guide

To set up a 1 GiB large web cache which will listen on port 8000, do the following:

```
[admin@MikroTik] ip web-proxy> set enabled=yes port=8000 max-cache-size=1048576
[admin@MikroTik] ip web-proxy> print
                   enabled: yes
               src-address: 0.0.0.0
                      port: 8000
                  hostname: proxy
         transparent-proxy: no
              parent-proxy: 0.0.0.0:0
       cache-administrator: webmaster
           max-object-size: 4096 KiB
               cache-drive: system
            max-cache-size: 1048576 KiB
        max-ram-cache-size: unlimited
                    status: rebuilding-cache
       reserved-for-cache: 9216 KiB
   reserved-for-ram-cache: 2048 KiB
[admin@MikroTik] ip web-proxy>
```

Remember to secure your proxy by preventing unauthorized access to it.

Specifications

Packages required: ***web-proxy***
License required: ***level3***
Home menu level: ***/ip web-proxy***
Standards and Technologies: ***HTTP/1.0***, ***HTTP/1.1***, ***FTP***
Hardware usage: *uses memory and disk space, if available (see description below)*

Related Documents

- ***Software Package Management***
- ***IP Addresses and ARP***
-

Description

Web proxy performs Internet object cache function by storing requested Internet objects, i.e., data available via HTTP and FTP protocols on a system positioned closer to the recipient than the site the data is originated from. Here 'closer' means increased path reliability, speed or both. Web browsers can then use the local proxy cache to speed up access and reduce bandwidth consumption.

When setting up Web proxy, make sure it serves only your clients, and is not misused as relay. Please read the security notice in the Access List Section!

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Note that it may be useful to have Web proxy running even with no cache when you want to use it as something like HTTP and FTP firewall (for example, denying access to mp3 files) or to redirect requests to external proxy transparently.

Setup

Home menu level: ***/ip web-proxy***

Property Description

cache-administrator (*text*; default: **webmaster**) - administrator's e-mail displayed on proxy error page

cache-drive (*system* | *name*; default: **system**) - specifies the target disk drive to be used for storing cached objects. You can use shell expansion to see available drives

enabled (yes | no; default: **no**) - specifies whether the web proxy is enabled

hostname (*text*; default: **proxy**) - hostname (DNS or IP address) of the web proxy

max-cache-size (*none* | *unlimited* | *integer*: 0..4294967295; default: **none**) - specifies the maximal disk cache size, measured in kibibytes

max-object-size (*integer*; default: **4096**) - objects larger than the size specified will not be saved on disk. The value is measured in kibibytes. If you wish to get a high bytes hit ratio, you should probably increase this (one 2 MiB object hit counts for 2048 1KiB hits). If you wish to increase speed more than your want to save bandwidth you should leave this low

max-ram-cache-size (*none* | *unlimited* | *integer*: 0..4294967295; default: **unlimited**) - specifies the maximal memory cache size, measured in kibibytes

parent-proxy (*IP address* | *port*; default: **0.0.0.0:0**) - specifies upper-level (parent) proxy

port (*port*; default: **3128**) - specifies the port(s) the web proxy will be listening on

reserved-for-cache (*read-only: integer*; default: **0**) - specifies allocated memory cache size, measured in kibibytes

reserved-for-ram-cache (*read-only: integer*; default: **2048**) - specifies allocated memory cache size, measured in kibibytes

src-address (*IP address*; default: **0.0.0.0**) - the web-proxy will use this address connecting to the parent proxy or web site.

- **0.0.0.0** - appropriate src-address will be automatically taken from the routing table

status (*read-only: text*; default: **stopped**) - display status information of the proxy server

- **stopped** - proxy is disabled and is not running
- **rebuilding-cache** - proxy is enabled and running, existing cache is being verified
- **running** - proxy is enabled and running
- **stopping** - proxy is shutting down (max 10s)
- **clearing-cache** - proxy is stopped, cache files are being removed
- **creating-cache** - proxy is stopped, cache directory structure is being created
- **dns-missing** - proxy is enabled, but not running because of unknown DNS server (you should specify it under /ip dns)
- **invalid-address** - proxy is enabled, but not running because of invalid address (you should

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

change address or port)

- **invalid-cache-administrator** - proxy is enabled, but not running because of invalid cache-administrator's e-mail address
- **invalid-hostname** - proxy is enabled, but not running because of invalid hostname (you should set a valid hostname value)
- **error-logged** - proxy is not running because of unknown error. This error is logged as System-Error. Please, send us this error and some description, how it happened
- **reserved-for-cache (integer)** - maximal cache size, that is accessible to web-proxy

transparent-proxy (yes | no; default: **no**) - specifies whether the proxy uses transparent mode or not

Notes

By default the proxy cache can use as much disk space as there is allocated for it. When the system allocates the space for the proxy cache, 1/7th of the total partition (disk) size is reserved for the system, but not less than 50MB. The rest is left for the proxy cache. The system RAM size is considered as well when allocating the cache size. The cache size is limited so, that there are at least 15MB of RAM per 1GB of cache plus 55MB of RAM is reserved for the system. **max-cache-size** is also taken in account, so the cache will not occupy more than it is specified in this property. The effective limit is calculated as a minimum of all three limits.

Considering the previous note, you should be aware that you will not be able to enable web proxy, if you have less than 60MB of RAM on your router

Expire time of cache entries can be different for each HTML page (specified in headers). But, if there is no such header, the entry will be considered fresh for not more than 72 hours.

The web proxy listens to all IP addresses that the router has in its IP address list.

Example

To enable the proxy on port 8080:

```
[admin@MikroTik] ip web-proxy> set enabled=yes port=8080
[admin@MikroTik] ip web-proxy> print
                   enabled: yes
               src-address: 0.0.0.0
                      port: 8080
                  hostname: proxy
         transparent-proxy: no
              parent-proxy: 0.0.0.0:0
       cache-administrator: webmaster
           max-object-size: 4096 KiB
               cache-drive: system
            max-cache-size: none
        max-ram-cache-size: unlimited
                    status: running
        reserved-for-cache: 0 KiB
    reserved-for-ram-cache: 2048 KiB
[admin@MikroTik] ip web-proxy>
```

Access List

Home menu level: ***/ip web-proxy access***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

Access list is implemented in the same way as MikroTik RouterOS firewall rules. Rules are processed from the top to the bottom. First matching rule specifies decision of what to do with this connection. There is a total of 6 classifiers that specify matching constraints. If none of these classifiers is specified, the particular rule will match every connection.

If connection is matched by a rule, **action** property of this rule specifies whether connection will be allowed or not. If the particular connection does not match any rule, it will be allowed.

By default, there exists one rule which prevents **connect** requests to ports other then **443** and **563**.

Property Description

action (*allow* | *deny*; default: **allow**) - specifies whether to pass or deny matched packets

dst-address (*IP address* | *netmask*) - destination address of the IP packet

dst-port (*port*) - a list or range of ports the packet is destined to

local-port (*port*) - specifies the port of the web proxy via which the packet was received. This value should match one of the ports web proxy is listening on.

method (*any* | *connect* | *delete* | *get* | *head* | *options* | *post* | *put* | *trace*) - HTTP method used in the request (see HTTP Methods section in the end of this document)

src-address (*IP address* | *netmask*) - source address of the IP packet

url (*wildcard*) - the URL of the HTTP request

Notes

There is one rule by default, that disallows **connect** method connections to ports other than **443** (https) and **563** (snews). **connect** method is a security hole that allows connections (transparent tunneling) to any computer using any protocol. It is used mostly by spammers, as they found it very convenient to use others' mail (SMTP) servers as anonymous mail relay to send spam over the Internet.

It is strongly recommended to deny all IP addresses except those behind the router as the proxy still may be used to access your internal-use-only (intranet) web servers. Also, consult examples in Firewall Manual on how to protect your router.

Wildcard property **url** matches a complete string (i.e., they will not match "example.com" if they are set to "example"). Available wildcards are '*' (match any number of any characters) and '?' (match any one character). Regular expressions are also accepted here, but if the property should be treated as a regular expression, it should start with a colon (':').

Small hits in using regular expressions:

- \\ symbol sequence is used to enter \ character in console
- **\.** pattern means **.** only (in regular expressions single dot in pattern means any symbol)
- to show that no symbols are allowed before the given pattern, we use ^ symbol at the beginning of the pattern
- to specify that no symbols are allowed after the given pattern, we use **$** symbol at the end of

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

the pattern

- to enter **[** or **]** symbols, you should escape them with backslash \.

Example

The default rule:

```
[admin@MikroTik] ip web-proxy access> print
Flags: X - disabled, I - invalid
 0   ;;; allow CONNECT only to SSL ports 443 [https] and 563 [snews]
     dst-port=!443,563 method=connect action=deny
[admin@MikroTik] ip web-proxy access>
```

To disallow download of .MP3 and .MPG files and FTP connections other than from the **10.0.0.1** server:

```
[admin@MikroTik] ip web-proxy access> add url=":\\.mp\[3g\]$" action=deny
[admin@MikroTik] ip web-proxy access> add src-address=10.0.0.1/32 action=allow
[admin@MikroTik] ip web-proxy access> add url="ftp://*" action=deny
[admin@MikroTik] ip web-proxy access> print
Flags: X - disabled, I - invalid
 0   ;;; allow CONNECT only to SSL ports 443 [https] and 563 [snews]
     dst-port=!443,563 method=connect action=deny

 1   url=":\.mp[3g]$" action=deny

 2   src-address=10.0.0.1/32 action=allow

 3   url="ftp://*" action=deny
[admin@MikroTik] ip web-proxy access>
```

Direct Access List

Home menu level: ***/ip web-proxy direct***

Description

If **parent-proxy** property is specified, it is possible to tell proxy server whether to try to pass the request to the parent proxy or to resolve it connecting to the requested server directly. Direct Access List is managed just like Proxy Access List described in the previous chapter except the **action** argument.

Property Description

action (*allow* | *deny*; default: **allow**) - specifies the action to perform on matched packets

- **allow** - always resolve matched requests directly bypassing the parent router
- **deny** - resolve matched requests through the parent proxy. If no one is specified this has the same effect as allow

dst-address (*IP address* | *netmask*) - destination address of the IP packet

dst-port (*port*) - a list or range of ports the packet is destined to

local-port (*port*) - specifies the port of the web proxy via which the packet was received. This value should match one of the ports web proxy is listening on.

method (*any* | *connect* | *delete* | *get* | *head* | *options* | *post* | *put* | *trace*) - HTTP method used in the request (see HTTP Methods section in the end of this document)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

src-address (*IP address* | *netmask*) - source address of the IP packet

url (*wildcard*) - the URL of the HTTP request

Notes

Unlike the access list, the direct proxy access list has default action equal to **deny**. It takes place when no rules are specified or a particular request did not match any rule.

Cache Management

Home menu level: ***/ip web-proxy cache***

Description

Cache access list specifies, which requests (domains, servers, pages) have to be cached locally by web proxy, and which not. This list is implemented exactly the same way as web proxy access list. Default action is to cache object (if no matching rule is found).

Property Description

action (*allow* | *deny*; default: **allow**) - specifies the action to perform on matched packets

- **allow** - cache objects from matched request
- **deny** - do not cache objects from matched request

dst-address (*IP address* | *netmask*) - destination address of the IP packet

dst-port (*port*) - a list or range of ports the packet is destined to

local-port (*port*) - specifies the port of the web proxy via which the packet was received. This value should match one of the ports web proxy is listening on.

method (*any* | *connect* | *delete* | *get* | *head* | *options* | *post* | *put* | *trace*) - HTTP method used in the request (see HTTP Methods section in the end of this document)

src-address (*IP address* | *netmask*) - source address of the IP packet

url (*wildcard*) - the URL of the HTTP request

Complementary Tools

Description

Web proxy has additional commands to handle non-system drive used for caching purposes and to recover the proxy from severe file system errors.

Command Description

check-drive - checks non-system cache drive for errors

clear-cache - deletes existing cache and creates new cache directories

format-drive - formats non-system cache drive and prepairs it for holding the cache

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Transparent Mode

Description

Transparent proxy feature performs request caching invisibly to the end-user. This way the user does not notice that his connection is being processed by the proxy and therefore does not need to perform any additional configuration of the software he is using.

Transparent proxy combined with bridge greatly simplifies deployment of web proxy in the existing infrastructure.

To enable the transparent mode, place a firewall rule in destination NAT, specifying which connections, *id est* traffic coming to which ports should be redirected to the proxy.

Notes

Only HTTP traffic is supported in transparent mode of the web proxy. HTTPS and FTP protocols are not going to work this way.

Example

To configure the router to transparently redirect all connections coming from **ether1** interface to port **80** to the web proxy listening on port **8080**, then add the following destination NAT rule:

```
[admin@MikroTik] > /ip firewall nat add in-interface=ether1 dst-port=80 \
\... protocol=tcp action=redirect to-ports=8080 chain=dstnat
[admin@MikroTik] > /ip firewall nat print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=dstnat protocol=tcp in-interface=ether1 dst-port=80 action=redirect
     to-ports=8080
[admin@MikroTik] >
```

Be aware, that you will not be able to access the router's web page after addition of the rule above unless you will change the port for the **www** service under **/ip service** submenu to a different value or explicitly exclude router's IP address from those to be matched, like:

It is assumed that the router's address is **1.1.1.1/32**.

HTTP Methods

Description

OPTIONS

This method is a request of information about the communication options available on the chain between the client and the server identified by the **Request-URI**. The method allows the client to determine the options and (or) the requirements associated with a resource without initiating any resource retrieval

GET

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

This method retrieves whatever information identified by the **Request-URI**. If the **Request-URI** refers to a data processing process than the response to the **GET** method should contain data produced by the process, not the source code of the process procedure(-s), unless the source is the result of the process.

The **GET** method can become a *conditional* **GET** if the request message includes an **If-Modified-Since, If-Unmodified-Since, If-Match, If-None-Match**, or **If-Range** header field. The conditional **GET** method is used to reduce the network traffic specifying that the transfer of the entity should occur only under circumstances described by conditional header field(-s).

The **GET** method can become a *partial* **GET** if the request message includes a **Range** header field. The partial **GET** method intends to reduce unnecessary network usage by requesting only parts of entities without transferring data already held by client.

The response to a **GET** request is cacheable if and only if it meets the requirements for HTTP caching.

HEAD

This method shares all features of **GET** method except that the server must not return a message-body in the response. This retrieves the metainformation of the entity implied by the request which leads to a wide usage of it for testing hypertext links for validity, accessibility, and recent modification.

The response to a **HEAD** request may be cacheable in the way that the information contained in the response may be used to update previously cached entity identified by that **Request-URI**.

POST

This method requests that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the **Request-URI**.

The actual action performed by the **POST** method is determined by the origin server and usually is **Request-URI** dependent.

Responses to **POST** method are not cacheable, unless the response includes appropriate **Cache-Control** or **Expires** header fields.

PUT

This method requests that the enclosed entity be stored under the supplied **Request-URI**. If another entity exists under specified **Request-URI**, the enclosed entity should be considered as updated (newer) version of that residing on the origin server. If the **Request-URI** is not pointing to an existing resource, the origin server should create a resource with that URI.

If the request passes through a cache and the **Request-URI** identifies one or more currently cached entities, those entries should be treated as stale. Responses to this method are not cacheable.

TRACE

This method invokes a remote, application-layer loop-back of the request message. The final recipient of the request should reflect the message received back to the client as the entity-body of a

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

200 (OK) response. The final recipient is either the origin server or the first proxy or gateway to receive a **Max-Forwards** value of **0** in the request. A **TRACE** request must not include an entity.

Responses to this method MUST NOT be cached.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Certificate Management

Document revision 2.3 (Fri Mar 05 13:58:17 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Description
Certificates
  Description
  Property Description
  Command Description
  Notes
  Example

General Information

Summary

SSL (Secure Socket Layer) is a security technology to ensure encrypted transactions over a public network. To protect the data, an encryption key should be negotiated. SSL protocol is using Certificates to negotiate a key for data encryption.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/certificate***
Standards and Technologies: ***SSLv2***, ***SSLv3***, ***TLS***
Hardware usage: *high CPU usage*

Description

SSL technology was first introduced by Netscape to ensure secure transactions between browsers and web servers. When a browser requests a secure web page (usually on TCP port 443), a web server first sends a Certificate, which contains a public key for the encryption key negotiation to take place. After the encryption key is negotiated, the web server will send the requested page encrypted using this key to the browser (and also the browser will be able to submit its data securely to the server)

SSL Certificate confirms the web server identity. The Certificate contains information about its holder (like DNS name and Country), issuer (the entity has signed the Certificate) and also the public key used to negotiate the encryption key. In order a Certificate to play its role, it should be signed by a third party (Certificate Authority) which both parties trust. Modern browsers that support SSL protocol have a list of the Certificate Authorities they trust (the most known and trusted CA is VeriSign, but that is not the only one)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

To use a Certificate (which contain a public key), server needs a private key. One of the keys is used for encryption, and the other - for decryption. It is important to understand, that both keys can encrypt and decrypt, but what is encrypted by one of them can be decrypted **only** by the another. Private key must be kept securely, so that nobody else can get it and use this certificate. Usually private key is encrypted with a passphrase.

Most trusted Certificate Authorities sell the service of signing Certificates (Certificates also have a finite validity term, so you will have to pay regularly). It is also possible to create a self-signed Certificate (you can create one on most UNIX/Linux boxes using openssl toolkit; all Root Certificate Authorities have self-signed Certificates), but if it is not present in a browser's database, the browser will pop up a security warning, saying that the Certificate is not trusted (note also that most browsers support importing custom Certificates to their databases).

Certificates

Home menu level: ***/certificate***

Description

MikroTik RouterOS can import Certificates for the SSL services it provides (only HotSpot for now). This submenu is used to manage Certificates for this services.

Property Description

name (*name*) - reference name

subject (*read-only: text*) - holder (subject) of the certificate

issuer (*read-only: text*) - issuer of the certificate

serial-number (*read-only: text*) - serial number of the certificate

invalid-before (*read-only: date*) - date the certificate is valid from

invalid-after (*read-only: date*) - date the certificate is valid until

ca (yes | no; default: **yes**) - whether the certificate is used for building or verifying certificate chains (as Certificate Authority)

Command Description

import - install new certificates

- **file-name** - import only this file (all files are searched for certificates by default)
- **passphrase** - passphrase for the found encrypted private key
- **certificates-imported** - how many new certificates were successfully imported
- **private-keys-imported** - how many private keys for existing certificates were successfully imported
- **files-imported** - how many files contained at least one item that was successfully imported
- **decryption-failures** - how many files could not be decrypted
- **keys-with-no-certificate** - how many public keys were successfully decrypted, but did not have matching certificate already installed

reset-certificate-cache - delete all cached decrypted public keys and rebuild the certificate cache

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

decrypt - decrypt and cache public keys

- **passphrase** - passphrase for the found encrypted private key
- **keys-decrypted** - how many keys were successfully decrypted and cached

create-certificate-request - creates an RSA certificate request to be signed by a Certificate Authority. After this, download both private key and certificate request files from the router. When you receive your signed certificate from the CA, upload it and the private key (that is made by this command) to a router and use /certificate import command to install it

- **certificate request file name** - name for the certificate request file (if it already exists, it will be overwritten). This is the original certificate that will be signed by the Certificate Authority
- **file name** - name of private key file. If such file does not exist, it will be created during the next step. Private key is used to encrypt the certificate
- **passphrase** - the passphrase that will be used to encrypt generated private key file. You must enter it twice to be sure you have not made any typing errors
- **rsa key bits** - number of bits for RSA (encryption) key. Longer keys take more time to generate. 4096 bit key takes about 30 seconds on Celeron 800 system to generate
- **country name** - (C) ISO two-character country code (e.g., LV for Latvia)
- **state or province name** - (ST) full name of state or province
- **locality name** - (L) locality (e.g. city) name
- **organization name** - (O) name of the organization or company
- **organization unit name** - (OU) organization unit name
- **common name** - (CN) the server's common name. For SSL web servers this must be the fully qualified domain name (FQDN) of the server that will use this certificate (like www.example.com). This is checked by web browsers
- **email address** - (Email) e-mail address of the person responsible for the certificate
- **challenge password** - the challenge password. It's use depends on your CA. It may be used to revoke this certificate
- **unstructured address** - unstructured address (like street address). Enter only if your CA accepts or requires it

Notes

Server certificates may have **ca** property set to **no**, but Certificate Authority certificates must have it set to **yes**

Certificates and encrypted private keys are imported from and exported to the router's FTP server. Public keys are not stored on a router in unencrypted form. Cached decrypted private keys are stored in encrypted form, using key that is derived from the router ID. Passphrases are not stored on router.

Configuration backup does not include cached decrypted private keys. After restoring backup all certificates with private keys must be decrypted again, using **decrypt** command with the correct passphrase.

No other certificate operations are possible while generating a key.

When making a certificate request, you may leave some of the fields empty. CA may reject your certificate request if some of these values are incorrect or missing, so please check what are the

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

requirements of your CA

Example

To import a certificate and the respective private key already uploaded on the router:

```
[admin@MikroTik] certificate> import
passphrase: xxxx
       certificates-imported: 1
       private-keys-imported: 1
              files-imported: 2
         decryption-failures: 0
    keys-with-no-certificate: 1
[admin@MikroTik] certificate> print
Flags: K - decrypted-private-key, Q - private-key, R - rsa, D - dsa
  0 QR name="cert1" subject=C=LV,ST=.,O=.,CN=cert.test.mt.lv
       issuer=C=LV,ST=.,O=.,CN=third serial-number="01"
       invalid-before=sep/17/2003 11:56:19 invalid-after=sep/16/2004 11:56:19
       ca=yes

[admin@MikroTik] certificate> decrypt
passphrase: xxxx
    keys-decrypted: 1
[admin@MikroTik] certificate> print
Flags: K - decrypted-private-key, Q - private-key, R - rsa, D - dsa
  0 KR name="cert1" subject=C=LV,ST=.,O=.,CN=cert.test.mt.lv
       issuer=C=LV,ST=.,O=.,CN=third serial-number="01"
       invalid-before=sep/17/2003 11:56:19 invalid-after=sep/16/2004 11:56:19
       ca=yes

[admin@MikroTik] certificate>
```

Now the certificate may be used by HotSpot servlet:

```
[admin@MikroTik] ip service> print
Flags: X - disabled, I - invalid
  #   NAME                                PORT  ADDRESS            CERTIFICATE
  0   telnet                              23    0.0.0.0/0
  1   ftp                                 21    0.0.0.0/0
  2   www                                 8081  0.0.0.0/0
  3   hotspot                             80    0.0.0.0/0
  4   ssh                                 22    0.0.0.0/0
  5   hotspot-ssl                         443   0.0.0.0/0          none

[admin@MikroTik] ip service> set hotspot-ssl certificate=
cert1  none
[admin@MikroTik] ip service> set hotspot-ssl certificate=cert1
[admin@MikroTik] ip service> print
Flags: X - disabled, I - invalid
  #   NAME                                PORT  ADDRESS            CERTIFICATE
  0   telnet                              23    0.0.0.0/0
  1   ftp                                 21    0.0.0.0/0
  2   www                                 8081  0.0.0.0/0
  3   hotspot                             80    0.0.0.0/0
  4   ssh                                 22    0.0.0.0/0
  5   hotspot-ssl                         443   0.0.0.0/0          cert1

[admin@MikroTik] ip service>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

DDNS Update Tool

Document revision 1.2 (Fri Mar 05 09:33:48 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
  Additional Documents
Dynamic DNS Update
  Property Description
  Notes
  Example

General Information

Summary

Dynamic DNS Update Tool gives a way to keep domain name pointing to dynamic IP address. It works by sending domain name system update request to name server, which has a zone to be updated. Secure DNS updates are also supported.

The DNS update tool supports only one algorithm - **hmac-md5**. It's the only proposed algorithm for signing DNS messages.

Specifications

Packages required: ***advanced-tools***
License required: ***level1***
Command name: ***/tool dns-update***
Standards and Technologies: ***Dynamic Updates in the DNS (RFC 2136)***, ***Secure DNS Dynamic Update (RFC 3007)***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***

Description

Dynamic DNS Update is a tool that should be manually run to update dynamic DNS server.

Note that you have to have a DNS server that supports DNS updates properly configured.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Additional Documents

- ***DNS related RFCs***

Dynamic DNS Update

Command name: ***/tool dns-update***

Property Description

address (*IP address*) - defines IP address associated with the domain name

dns-server (*IP address*) - DNS server to send update to

key (*text*; default: **""**) - authorization key (password of a kind) to access the server

key-name (*text*; default: **""**) - authorization key name (username of a kind) to access the server

name (*text*) - name to attach with the IP address

ttl (*integer*; default: **0**) - time to live for the item (in seconds)

zone (*text*) - DNS zone where to update the domain name in

Notes

Example

To tell **23.34.45.56** DNS server to (re)associate **mydomain** name in the **myzone.com** zone with **68.42.14.4** IP address specifying that the name of the key is **dns-update-key** and the actual key is **update**:

```
[admin@MikroTik] tool> dns-update dns-server=23.34.45.56 name=mydomain \
\... zone=myzone.com address=68.42.14.4 key-name=dns-update-key key=update
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

GPS Synchronization

Document revision 2.0 (Fri Mar 05 08:56:37 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
  Additional Documents
Synchronizing with a GPS Receiver
  Property Description
  Notes
  Example
GPS Monitoring
  Description
  Property Description
  Example

General Information

Summary

Global Positioning System (GPS) receiver can be used by MikroTik RouterOS to get the precise location and time (which may be used as NTP time source)

Specifications

Packages required: ***gps***
License required: ***level1***
Home menu level: ***/system gps***
Standards and Technologies: ***GPS, NMEA 0183, Simple Text Output Protocol***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***NTP (Network Time Protocol)***

Description

Global Positioning System (GPS) is used for determining precise location of a GPS receiver. There are two types of GPS service:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- Precise Positioning Service (PPS) that is used only by U. S. and Allied military, certain U. S. Government agencies, and selected civil users specifically approved by the U. S. Government. Its accuracy is 22m horizontally, 27.7m vertically and 200ns of time
- Standard Positioning Service (SPS) can be used by civil users worldwide without charge or restrictions except that SPS accuracy is intentionally degradated to 100m horizontally, 156m vertically and 340ns of time

GPS system is based on 24 satellites rotating on 6 different orbital planes with 12h orbital period. It makes that at least 5, but usually 6 or more satellites are visible at any time anywhere on the Earth. GPS receiver calculates more or less precise position (latitude, longitude and altitude) and time based on signals received from 4 satellites (three are used to determine position and fourth is used to correct time), which are broadcasting their current positions and UTC time.

MikroTik RouterOS can communicate with many GPS receivers which are able to send the positioning and time via asynchronous serial line using NMEA 0183, NMEA/RTCM or Simple Text Output Protocol. Note that you might need to configure the router's serial port in order to work with your device. For example, many GPS receivers work on 4800bit/s bitrate, to the same whould be set in the **/port** menu for the respective serial port.

Precise time is mainly intended to be used by built-in NTP server, which can use it as a time source without any additional configuration if GPS is configured to set system time.

Additional Documents

- ***Global Positioning System - How it Works***

Synchronizing with a GPS Receiver

Home menu level: ***/system gps***

Property Description

enabled (*yes* | *no*) - whether the router will communicate with a GPS receiver or not

port (*name*) - the port that will be used to communicate with a GPS receiver

set-system-time (*time*) - whether to set the system time to the value received from a GPS receiver or not

Notes

If you are synchronizing system time with a GPS device, you should correctly choose time zone if it is different from GMT as satellites are broadcasting GMT (a.k.a. UTC) time.

Example

To enable GPS communication through serial0 port:

```
[admin@MikroTik] system gps> print
             enabled: no
                port: (unknown)
    set-system-time: yes
[admin@MikroTik] system gps> set enabled=yes port=serial0
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] system gps> print
            enabled: yes
               port: serial0
    set-system-time: yes
[admin@MikroTik] system gps>
```

GPS Monitoring

Home menu level: ***/system gps monitor***

Description

This command is used for monitoring the data received from a GPS receiver.

Property Description

date-and-time (*read-only: text*) - date and time received from GPS server

longitude (*read-only: text*) - longitude of the current location

latitude (*read-only: text*) - latitude of the current location

altitude (*read-only: text*) - altitude of the current location

speed (*read-only: text*) - mean velocity

valid (*read-only: yes | no*) - whether the received information is valid or not (e.g. you can set a GPS receiver to the demo mode to test the connection, in which case you will receive information, but it will not be valid)

Example

```
[admin@MikroTik] system gps> monitor
    date-and-time: jul/23/2003 12:25:00
        longitude: "E 24 8' 17''"
         latitude: "N 56 59' 22''"
         altitude: "-127.406400m"
            speed: "0.001600 km/h"
            valid: yes

[admin@MikroTik] system gps>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

LCD Management

Document revision 2.2 (Thu Mar 03 11:48:21 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
Configuring the LCD's Settings
  Property Description
  Example
LCD Information Display Configuration
  Description
  Property Description
  Notes
  Example
LCD Troubleshooting
  Description

General Information

Summary

LCDs are used to display system information.

The MikroTik RouterOS supports the following LCD hardware:

- Crystalfontz (***http://www.crystalfontz.com***) Intelligent Serial LCD Module 632 (16x2 characters) and 634 (20x4 characters)
- Powertip (***http://www.powertip.com.tw***) PC1602 (16x2 characters), PC1604 (16x4 characters), PC2002 (20x2 characters), PC2004 (20x4 characters), PC2402 (24x2 characters) and PC2404 (24x4 characters)

Specifications

Packages required: ***lcd***
License required: ***level1***
Home menu level: ***/system lcd***
Standards and Technologies: ***None***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

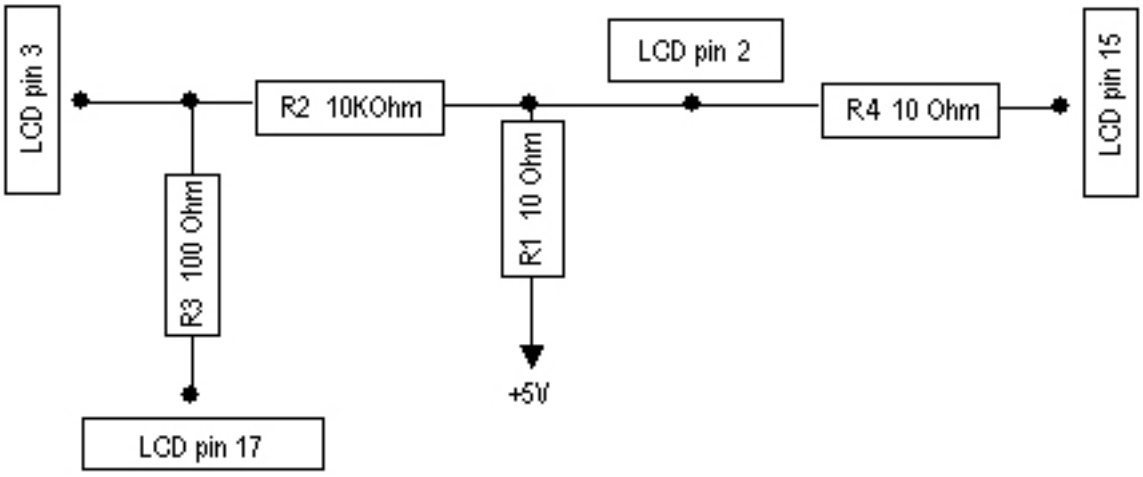
Description

How to Connect PowerTip LCD to a Parallel Port

Data signals are connected that way:

DB25m	Signal	LCD Panel
1	Enable (Strobe)	6
2	Data 0	7
3	Data 1	8
4	Data 2	9
5	Data 3	10
6	Data 4	11
7	Data 5	12
8	Data 6	13
9	Data 7	14
14	Register Select	4
18-25, GND	Ground	1, 5, 16

Powering:



As there are only 16 pins for the PC1602 modules, you need not connect power to the 17th pin.

GND and +5V can be taken from computer's internal power supply (use black wire for GND and red wire for +5V)

WARNING! Be very careful connecting power supply. We do not recommend using external power supplies. In no event shall MikroTik liable for any hardware damages.

Note that there are some PowerTip PC2404A modules that have different pin-out. Compare:

- ***From www.powertip.com.tw (probably newer one)***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***From www.actron.de (probably older one)***

Some LCDs may be connected without resistors:

DB25m	Signal	LCD Panel
18-25, GND	**Ground**	**1, 3, 4, 16**
+5V	**Power**	**2, 15**

Crystalfontz LCD Installation Notes

Before connecting the LCD, please check the availability of ports, their configuration, and free the desired port resource, if required:

```
[admin@MikroTik] port> print
  # NAME                      USED-BY                         BAUD-RATE
  0 serial0                      Serial Console                   9600
  1 serial1                                                           9600
[admin@MikroTik] port>
```

Configuring the LCD's Settings

Home menu level: ***/system lcd***

Property Description

contrast (*integer*: 0..255; default: **0**) - contrast setting, sent to the LCD, if it supports contrast regulations

enabled (*yes* | *no*; default: **no**) - turns the LCD on or off

port (*name* | *parallel*; default: **parallel**) - name of the port where the LCD is connected. May be either one of the serial ports, or the first parallel

type (*16x2* | *16x4* | *20x2* | *20x4* | *24x2* | *24x4*; default: **24x4**) - sets the size of the LCD

Example

To enable Powertip parallel port LCD:

```
[admin@MikroTik] system lcd> print
   enabled: no
      type: 24x4
      port: parallel
  contrast: 0
[admin@MikroTik] system lcd> set enabled=yes
[admin@MikroTik] system lcd> print
   enabled: yes
      type: 24x4
      port: parallel
  contrast: 0
[admin@MikroTik] system lcd>
```

To enable Crystalfontz serial LCD on **serial1**:

```
[admin@MikroTik] system lcd> set port=serial1
[admin@MikroTik] system lcd> print
   enabled: yes
      type: 24x4
      port: serial1
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  contrast: 0
[admin@MikroTik] system lcd>
```

LCD Information Display Configuration

Home menu level: ***/system lcd page***

Description

The submenu is used for configuring LCD information display: what pages and how long will be shown.

Property Description

description (*read-only: text*) - page description

display-time (*time*; default: **5s**) - how long to display the page

Notes

You cannot neither add your own pages (they are created dynamically depending on the configuration) nor change pages' description.

Example

To enable displaying all the pages:

```
[admin@MikroTik] system lcd page> print
Flags: X - disabled
  #   DISPLAY-TIME    DESCRIPTION
  0 X 5s            System date and time
  1 X 5s            System resources - cpu and memory load
  2 X 5s            System uptime
  3 X 5s            Aggregate traffic in packets/sec
  4 X 5s            Aggregate traffic in bits/sec
  5 X 5s            Software version and build info
  6 X 5s            ether1
  7 X 5s            prism1
[admin@MikroTik] system lcd page> enable [find]
[admin@MikroTik] system lcd page> print
Flags: X - disabled
  #   DISPLAY-TIME    DESCRIPTION
  0   5s            System date and time
  1   5s            System resources - cpu and memory load
  2   5s            System uptime
  3   5s            Aggregate traffic in packets/sec
  4   5s            Aggregate traffic in bits/sec
  5   5s            Software version and build info
  6   5s            ether1
  7   5s            prism1
[admin@MikroTik] system lcd page>
```

To set "System date and time" page to be displayed for 10 seconds:

```
[admin@MikroTik] system lcd page> set 0 display-time=10s
[admin@MikroTik] system lcd page> print
Flags: X - disabled
  #   DISPLAY-TIME    DESCRIPTION
  0   10s            System date and time
  1   5s            System resources - cpu and memory load
  2   5s            System uptime
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  3   5s            Aggregate traffic in packets/sec
  4   5s            Aggregate traffic in bits/sec
  5   5s            Software version and build info
  6   5s            ether1
  7   5s            prism1
[admin@MikroTik] system lcd page>
```

LCD Troubleshooting

Description

LCD doesn't work, cannot be enabled by the '/system lcd set enabled=yes' command.

Probably the selected serial port is used by PPP client or server, or by the serial console. Check the availability and use of the ports by examining the output of the **/port print** command. Alternatively, select another port for connecting the LCD, or free up the desired port by disabling the related resource

LCD doesn't work, does not show any information.

Probably none of the information display items have been enabled. Use the */system lcd page set* command to enable the display.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

MNDP

Document revision 1.4 (Fri Mar 05 08:36:57 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
Setup
  Property Description
  Example
Neighbour List
  Description
  Property Description
  Example

General Information

Summary

The MikroTik Neighbor Discovery Protocol (MNDP) eases network configuration and management by enabling each MikroTik router to discover other connected MikroTik routers and learn information about the system along with features which are enabled. The MikroTik routers can automatically use learned information to set up some features with minimal or no configuration.

MNDP features:

- works on IP level connections
- works on all non-dynamic interfaces
- distributes basic information on the software version
- distributes information on configured features that should interoperate with other MikroTik routers

MikroTik RouterOS is able to discover both MNDP and CDP (Cisco Discovery Protocol) devices.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip neighbor***
Standards and Technologies: ***MNDP***
Hardware usage: ***Not significant***

Related Documents

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***Package Management***
- ***M3P***

Description

MNDP basic function is to assist with automatic configuration of features that are only available between MikroTik routers. Currently this is used for the 'Packet Packer' feature. The 'Packet Packer' may be enabled on a per interface basis. The MNDP protocol will then keep information about what routers have enabled the 'unpack' feature and the 'Packet Packer' will be used for traffic between these routers.

Specific features

- works on interfaces that support IP protocol and have at least one IP address and on all ethernet-like interfaces even without IP addresses
- is enabled by default for all new Ethernet-like interfaces -- Ethernet, wireless, EoIP, IPIP tunnels, PPTP-static-server
- when older versions on the RouterOS are upgraded from a version without discovery to a version with discovery, current Ethernet like interfaces will not be automatically enabled for MNDP
- uses UDP protocol port 5678
- a UDP packet with router info is broadcasted over the interface every 60 seconds
- every 30 seconds, the router checks if some of the neighbor entries are not stale
- if no info is received from a neighbor for more than 180 seconds the neighbor information is discarded

Setup

Home menu level: ***/ip neighbor discovery***

Property Description

name (*read-only: name*) - interface name for reference

discover (yes | no; default: **yes**) - specifies whether the neighbour discovery is enabled or not

Example

To disable MNDP protocol on Public interface:

```
[admin@MikroTik] ip neighbor discovery> set Public discover=no
[admin@MikroTik] ip neighbor discovery> print
  # NAME      DISCOVER
  0 Public    no
  1 Local     yes
```

Neighbour List

Home menu level: ***/ip neigbor***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

This submenu allows you to see the list of neighbours discovered

Property Description

interface (*read-only: name*) - local interface name the neighbour is connected to

address (*read-only: IP address*) - IP address of the neighbour router

mac-address (*read-only: MAC address*) - MAC address of the neighbour router

identity (*read-only: text*) - identity of the neighbour router

version (*read-only: text*) - operating system or firmware version of the neighbour router

unpack (*read-only: none* | *simple* | *compress-headers* | *compress-all*) - identifies if the interface of the neighbour router is unpacking packets packed with M3P

platform (*read-only: text*) - hardware/software platworm type of neighbour router

age (*read-only: time*) - specifies the record's age in seconds (time from last update)

Example

To view the table of discovered neighbours:

```
[admin@MikroTik] ip neighbor> pri
 # INTERFACE ADDRESS         MAC-ADDRESS       IDENTITY   VERSION
 0 ether2    10.1.0.113      00:0C:42:00:02:06 ID         2.9beta5
 1 ether2    1.1.1.3         00:0C:42:03:02:ED MikroTik   2.9beta5
[admin@MikroTik] ip neighbor>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

NTP (Network Time Protocol)

Document revision NaN (Mon Jul 19 07:25:46 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
Client
  Property Description
  Example
Server
  Property Description
  Notes
  Example
Time Zone
  Notes
  Example

General Information

Summary

NTP protocol allows synchronizing time among computers in network. It is good if there is an internet connection available and local NTP server is synchronized to correct time source. List of publec NTP servers is available at ***http://www.eecis.udel.edu/~mills/ntp/servers.html***

Specifications

Packages required: ***ntp***
License required: ***level1***
Home menu level: ***/system ntp***
Standards and Technologies: ***NTP version 3 (RFC 1305)***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***IP Addresses and ARP***

Description

Network Time Protocol (NTP) is used to synchronize time with some NTP servers in a network.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

MikroTik RouterOS provides both - NTP client and NTP server.

NTP server listens on UDP port 123

NTP client synchronizes local clock with some other time source (NTP server). There are 4 modes in which NTP client can operate at:

- **unicast** (Client/Server) mode - NTP client connects to specified NTP server. IP address of NTP server must be set in ntp-server and/or second-ntp-server parameters. At first client synchronizes to NTP server. Afterwards client periodically (64..1024s) sends time requests to NTP server. Unicast mode is the only one which uses ntp-server and second-ntp-server parameters.
- **broadcast** mode - NTP client listens for broadcast messages sent by NTP server. After receiving first broadcast message, client synchronizes local clock using unicast mode, and afterwards does not send any packets to that NTP server. It uses received broadcast messages to adjust local clock.
- **multicast** mode - acts the same as broadcast mode, only instead of broadcast messages (IP address 255.255.255.255) multicast messages are received (IP address 224.0.1.1).
- **manycast** mode - actually is unicast mode only with unknown IP address of NTP server. To discover NTP server, client sends multicast message (IP 239.192.1.1). If NTP server is configured to listen for these multicast messages (manycast mode is enabled), it replies. After client receives reply, it enters unicast mode and synchronizes to that NTP server. But in parallel client continues to look for more NTP servers by sending multicast messages periodically.

Client

Home menu level: ***/system ntp client***

Property Description

enabled (*yes* | *no*; default: **no**) - whether the NTP client is enabled or not

mode (*unicast* | *broadcast* | *multicast* | *manycast*; default: **unicast**) - NTP client mode

primary-ntp (*IP address*; default: **0.0.0.0**) - specifies IP address of the primary NTP server

secondary-ntp (*IP address*; default: **0.0.0.0**) - specifies IP address of the secondary NTP server

status (*read-only: text*) - status of the NTP client:

- **stopped** - NTP is not running (NTP is disabled)
- **error** - there was some internal error starting NTP service (please, try to restart (disable and enable) NTP service)
- **started** - NTP client service is started, but NTP server is not found, yet
- **failed** - NTP server sent invalid response to our NTP client (NTP server is not synchronized to some other time source)
- **reached** - NTP server contacted. Comparing local clock to NTP server's clock (duration of this phase is approximately 30s)
- **timeset** - local time changed to NTP server's time (duration of this phase is approximately 30s)
- **synchronized** - local clock is synchronized to NTP server's clock. NTP server is activated

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **using-local-clock** - using local clock as time source (server enabled while client disabled)

Example

To enable the NTP client to synchronize with the **159.148.60.2** server:

```
[admin@MikroTik] system ntp client> set enabled=yes primary-ntp=159.148.60.2
[admin@MikroTik] system ntp client> print
           enabled: yes
              mode: unicast
       primary-ntp: 159.148.60.2
     secondary-ntp: 0.0.0.0
            status: synchronized
[admin@MikroTik] system ntp client>
```

Server

Home menu level: ***/system ntp server***

Property Description

broadcast (*yes* | *no*; default: **no**) - whether NTP broadcast message is sent to 255.255.255.255 every 64s

enabled (*yes* | *no*; default: **no**) - whether the NTP server is enabled

manycast (*yes* | *no*; default: **yes**) - whether NTP server listens for multicast messages sent to 239.192.1.1 and responds to them

multicast (*yes* | *no*; default: **no**) - whether NTP multicast message is sent to 224.0.1.1 every 64s

Notes

NTP server activities only when local NTP client is in **synchronized** or **using-local-clock** mode.

If NTP server is disabled, all NTP requests are ignored.

If NTP server is enabled, all individual time requests are answered.

CAUTION! Using **broadcast**, **multicast** and **manycast** modes is dangerous! Intruder (or simple user) can set up his own NTP server. If this new server will be chosen as time source for your server, it will be possible for this user to change time on your server at his will.

Example

To enable NTP server to answer unicast requests only:

```
[admin@MikroTik] system ntp server> set manycast=no enabled=yes
[admin@MikroTik] system ntp server> print
       enabled: yes
     broadcast: no
     multicast: no
      manycast: no
[admin@MikroTik] system ntp server>
```

Time Zone

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Home menu level: ***/system clock***

Notes

NTP changes local clock to UTC (GMT) time by default.

Example

Time zone is specified as a difference between local time and GMT time. For example, if GMT time is 10:24:40, but correct local time is 12:24:40, then time-zone has to be set to +2 hour:

```
[admin@MikroTik] system clock> print
         time: dec/24/2003 10:24:40
    time-zone: +00:00
[admin@MikroTik] system clock> set time-zone=+02:00
[admin@MikroTik] system clock> print
         time: dec/24/2003 12:24:42
    time-zone: +02:00
[admin@MikroTik] system clock>
```

If local time is before GMT time, time-zone value will be negative. For example, if GMT is 18:00:00, but correct local time is 15:00:00, time-zone has to be set to -3 hours:

```
[admin@MikroTik] system clock> set time-zone=-3
[admin@MikroTik] system clock> print
         time: sep/24/2004 08:13:28
    time-zone: -03:00
[admin@MikroTik] system clock>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

RouterBoard-specific functions

Document revision 3 (Wed Jul 06 11:26:35 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
BIOS upgrading
  Description
  Property Description
  Command Description
  Example
BIOS Configuration
  Description
  Property Description
  Example
System Health Monitoring
  Description
  Property Description
  Notes
  Example
LED Management or RB200
  Description
  Property Description
  Notes
  Example
LED Management on RB500
  Description
  Property Description
Fan voltage control
  Description
  Property Description
Console Reset Jumper
  Description

General Information

Summary

There are some features used to configure specific functions exist only in RouterBOARD series embedded routers:

- BIOS upgrading
- BIOS configuration

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- Health monitoring (RouterBOARD 200 series only)
- LED control (may be used in scripting)
- Fan voltage control (on/off) (RouterBOARD 200 series only)
- Console reset jumper (RouterBOARD 200 series only)

Specifications

Packages required: ***routerboard***
License required: ***level1***
Home menu level: ***/system routerboard, /system health***
Hardware usage: *works only on RouterBOARD platform*

BIOS upgrading

Home menu level: ***/system routerboard***

Description

The BIOS is needed to recognize all the hardware and boot the system up. Newer BIOS versions might have support for more hardware, so it's generally a good idea to upgrade the BIOS once a newer version is available.

The newest versions of BIOS firmware is included in the newest **routerboard** software package. BIOS firmware may also be uploaded to router's FTP server (the file is called **wlb-bios.rom**). This way, for example, BIOS firmware may be transferred from one router to an another.

Property Description

current-firmware (*read-only: text*) - the version and build date of the BIOS already flashed

model (*read-only: text*) - RouterBOARD model

routerboard (*read-only:* yes | no) - whether the motherboard has been detected as a RouterBOARD

serial-number (*read-only: text*) - RouterBOARD serial number

upgrade-firmware (*read-only: text*) - the version and build date of the BIOS that is available for flashing

Command Description

upgrade - write the uploaded firmware to the BIOS (asks confirmation, and then reboots the router)

Example

To check the current and available firmware version numbers:

```
[admin@MikroTik] system routerboard> print
         routerboard: yes
               model: 230
       serial-number: 8524983
    current-firmware: 1.3.4beta7 (Nov/12/2004 17:12:58)
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
    upgrade-firmware: 1.3.4beta7 (Nov/16/2004 17:02:35)
[admin@MikroTik] system routerboard>
```

To upgrade the BIOS version:

```
[admin@MikroTik] > system routerboard upgrade
Firmware upgrade requires reboot of the router. Continue? [y/n] y
Firmware upgrade can take up to 20s. Do NOT turn off the power!
```

BIOS Configuration

Home menu level: ***/system routerboard bios***

Description

In addition to BIOS own setup possibilities, it is possible to configure BIOS parameters in RouterOS condole

Property Description

baud-rate (*1200* | *2400* | *4800* | *9600* | *19200* | *38400* | *57600* | *115200*; default: **9600**) - initian bitrate of the onboard serial port

beep-on-boot (yes | no; default: **yes**) - whether to beep during boot procedure (to indicate that it has succeeded)

boot-delay (*time*: 0s..10s; default: **1s**) - how much time to wait for a key storke while booting

boot-device (*etherboot-ide* | *etherboot-only* | *ide-only* | *try-etherboot-once*) - specifies from which device the RouterBoard will boot

- **etherboot-ide** - boot from etherboot, if it fails, boot from ide
- **etherboot-only** - boot only from etherboot
- **ide-only** - boot only from ide
- **try-etherboot-once** - boot from etherboot once, then returns to previous settings

cpu-mode (*power-save* | *regular*; default: **power-save**) - whether to enter CPU suspend mode in HTL instruction. Most OSs use HLT instruction during CPU idle cycle. When CPU is in suspend mode, it consumes less power, but in low-temperatire conditions it is recommended to choose regular mode, so that overall system temperature would be greater

debug-level (*none* | *low* | *high*) - BIOS output debug level

- **none** - no debugging output
- **low** - show only some debugging information
- **high** - show all debugging information about the boot process

enter-setup-on (*any-key* | *delete-key*; default: **any-key**) - which key will cause the BIOS to enter configuration mode during boot delay. Note that in some serial terminal programs, it is impossible to use Delete key to enter the setup - in this case it might be possible to do this with the Backspace key

etherboot-timeot (*time*; default: **1m**) - how much time to wait for booting from ethernet

memory-settings (*fail-safe* | *optimal*; default: **auto**) - SDRAM memory speed

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **optimal** - the BIOS tries to determine the correct memory settings by itself
- **fail-safe** - use if you have memory-related errors (generally random, not reproducible errors and freezes). In this case, minimal timing parametrs are used, so most memory modules will work reliably

memory-test (yes | no; default: **no**) - whether to testall the RAM during boot procedure. Regardless of the choice, hte first megabyte of the RAM will be tested anyway. Enabling this option may cause longer boot process

pci-backoff (*enabled* | *disabled*; default: **enabled**) - when enabled, external PCI masters can access system memory even when a CPU cycle has been retried. If you are experiencing uncommon problems with PCI/PCMCIA/CardBUS interfaces (including RouterBOARD is rebooting or hanging up once in a while), try to disable it

vga-to-serial (yes | no; default: **yes**) - whether to map VGA output to the serial console. Should be enabled if working via serial terminal (gives much more output)

Example

To set high debug level with RAM test:

```
[admin@MikroTik] system routerboard bios> print
            baud-rate: 9600
          debug-level: low
           boot-delay: 00:00:01
       enter-setup-on: any-key
         beep-on-boot: yes
          boot-device: ide-only
    etherboot-timeout: 00:01:00
        vga-to-serial: yes
      memory-settings: optimal
          memory-test: no
             cpu-mode: power-save
          pci-backoff: enabled
[admin@MikroTik] system routerboard bios> set debug-level=high
[admin@MikroTik] system routerboard bios> print
            baud-rate: 9600
          debug-level: high
           boot-delay: 00:00:01
       enter-setup-on: any-key
         beep-on-boot: yes
          boot-device: ide-only
    etherboot-timeout: 00:01:00
        vga-to-serial: yes
      memory-settings: optimal
          memory-test: no
             cpu-mode: power-save
          pci-backoff: enabled
[admin@MikroTik] system routerboard bios>
```

System Health Monitoring

Home menu level: ***/system health***

Description

LM87 health controller chip provides some measurments of temperature and voltage on RouterBOARD 200 series computers. Information becomes available not sooner than 2 minutes after boot up. It is not available if LM87 chip is not detected successfully. All values are 10 second

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

averages, with short peak values ignored as likely read errors

Property Description

3.3v - +3.3V power line voltage

5v - +5V power line voltage

board-temp - temperature of the PCI area

core - CPU core voltage

cpu-temp - temperature of the CPU area

lm87-temp - temperature of the LM87 chip

state (*read-only: enabled* | *disabled*; default: **disabled**) - the current state of health monitoring (whether it is enabled or not)

state-after-reboot (*enabled* | *disabled*; default: **disabled**) - the state of the health monitor after the reboot

Notes

You cannot change state on the fly, just control, whether the health control will be enabled after reboot

All themperature values are in Celsius degrees

Example

To check system health:

```
[admin@MikroTik] system health> print
                    core: 1.32
                    3.3v: 3.26
                      5v: 4.97
              lm87-temp: -0.9
               cpu-temp: -0.9
             board-temp: -0.9
                  state: enabled
    state-after-reboot: enabled
[admin@MikroTik] system routerboard health>
```

LED Management or RB200

Command name: ***:led***

Description

The four user LEDs of the RouterBOARD 200 series can be controlled from user-space scripts.

Property Description

led1 (yes | no; default: **no**) - whether the LED1 is on

led2 (yes | no; default: **no**) - whether the LED2 is on

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

led3 (yes | no; default: **no**) - whether the LED3 is on

led4 (yes | no; default: **no**) - whether the LED3 is on

length (*time*; default: **0s**) - how long to hold the given combination

- **0s** - no limit

Notes

The command does not imply a pause in execution. It works asynchronously, allowing execution to continue just after the command was entered, not waiting for LEDs to switch off.

After the given time (**length** property) the LEDs will return to the default (off) condition.

Any new **:led** command overrides the the previous state and resets the LED state after the **length** time interval.

Example

To turn LED1 on for a minute:

```
[admin@MikroTik] > :led led1=yes length=1m
[admin@MikroTik] >
```

LED Management on RB500

Command name: ***/blink***

Description

It is possible to blink with the only user LED (the red one, near the blue power LED) of ROuterBOARD 500 series boards

Property Description

duration (*time*; default: **10s**) - how long to flash the red LED

Fan voltage control

Command name: ***/system routerboard fan-control***

Description

On RouterBOARD 200 series you can control, whether the J11 fan 5V voltage output is enabled. This feature will only work with newest BIOS versions. This is useful in scripts to control some devices attached to the J11 connector.

Property Description

length (*time*; default: **0**) - how long to hold the set state value, and then return to the prevoius state

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **0** - leave the state in the set mode until restart

state (yes | no) - whether to enable the 5V output on pins 1-2 of the J11 header

Console Reset Jumper

Description

The J16 jumper on the RouterBOARD 200 may be used as serial console reset pin. If it held short for at least 10 seconds, then:

- Serial console configuration is reset
- Serial port that serial console will pick by default (usually serial0) is set to 9600 baud 8 bit 1 stop bit no parity (default settings after installation)
- Special flag that prevents any other program except serial console to acquire this port is set
- Router is rebooted

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Support Output File

Document revision 2.1.0 (Wed Mar 03 16:11:16 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
Generating Support Output File
  Example

General Information

Summary

The support file is used for debugging MikroTik RouterOS and to solve the support questions faster. All MikroTik Router information is saved in a binary file, which is stored on the router and can be downloaded from the router using ftp.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/system***
Hardware usage: ***Not significant***

Generating Support Output File

Command name: ***/system sup-output***

Example

To make a Support Output File:

```
[admin@MikroTik] > system sup-output
creating supout.rif file, might take a while
.................
Done!
[admin@MikroTik] >
```

To see the files stored on the router:

```
[admin@MikroTik] > file print
 # NAME                          TYPE         SIZE       CREATION-TIME
 0 supout.rif                    unknown      108787     dec/24/2003 10:12:38
[admin@MikroTik] >
```

Connect to the router using FTP and download the supout.rif file using BINARY file transfer mode. Send the supout.rif file to MikroTik Support ***support@mikrotik.com*** with detailed description of the problem.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

System Resource Management

Document revision 2.1 (Tue Sep 27 17:38:19 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
System Resource
  Notes
  Example
IRQ Usage Monitor
  Description
  Example
IO Port Usage Monitor
  Description
  Example
USB Port Information
  Description
  Property Description
  Example
PCI Information
  Property Description
  Example
Reboot
  Description
  Notes
  Example
Shutdown
  Description
  Notes
  Example
Router Identity
  Description
  Example
Date and Time
  Property Description
  Notes
  Example
Configuration Change History
  Description
  Command Description
  Notes
  Example
System Note

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description
Property Description
Notes

General Information

Summary

MikroTik RouterOS offers several features for monitoring and managing the system resources.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/system***
Standards and Technologies: ***None***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***NTP (Network Time Protocol)***

System Resource

Home menu level: ***/system resource***

Notes

In **monitor** command priotout the values for cpu usage and free memory are in percentage and kilobytes, respectively.

Example

To view the basic system resource status:

```
[admin@MikroTik] system resource> print
                     uptime: 04:32:41
                free-memory: 46488 kB
               total-memory: 62672 kB
                      model: RouterBOARD 230
                        cpu: Geode
                  cpu-load: 0
             free-hdd-space: 35873 kB
            total-hdd-space: 61972 kB
    write-sect-since-reboot: 2678
           write-sect-total: 408130
[admin@MikroTik] system resource>
```

To view the current system CPU usage and free memory:

```
[admin@MikroTik] > system resource monitor
       cpu-used: 0
    free-memory: 115676
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] >
```

IRQ Usage Monitor

Command name: ***/system resource irq print***

Description

IRQ usage shows which IRQ (Interrupt requests) are currently used by hardware.

Example

```
[admin@MikroTik] > system resource irq print
Flags: U - unused
   IRQ OWNER
   1   keyboard
   2   APIC
 U 3
   4   serial port
   5   [Ricoh Co Ltd RL5c476 II (#2)]
 U 6
 U 7
 U 8
 U 9
 U 10
   11  ether1
   12  [Ricoh Co Ltd RL5c476 II]
 U 13
   14  IDE 1
[admin@MikroTik] >
```

IO Port Usage Monitor

Command name: ***/system resource io print***

Description

IO usage shows which IO (Input/Output) ports are currently used by hardware.

Example

```
[admin@MikroTik] > system resource io print
 PORT-RANGE        OWNER
 0x20-0x3F         APIC
 0x40-0x5F         timer
 0x60-0x6F         keyboard
 0x80-0x8F         DMA
 0xA0-0xBF         APIC
 0xC0-0xDF         DMA
 0xF0-0xFF         FPU
 0x1F0-0x1F7       IDE 1
 0x2F8-0x2FF       serial port
 0x3C0-0x3DF       VGA
 0x3F6-0x3F6       IDE 1
 0x3F8-0x3FF       serial port
 0xCF8-0xCFF       [PCI conf1]
 0x4000-0x40FF     [PCI CardBus #03]
 0x4400-0x44FF     [PCI CardBus #03]
 0x4800-0x48FF     [PCI CardBus #04]
 0x4C00-0x4CFF     [PCI CardBus #04]
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  0x5000-0x500F     [Intel Corp. 82801BA/BAM SMBus]
  0xC000-0xC0FF     [Realtek Semiconductor Co., Ltd. RTL-8139/8139C/8139C+]
  0xC000-0xC0FF     [8139too]
  0xC400-0xC407     [Cologne Chip Designs GmbH ISDN network controller [HFC-PCI]
  0xC800-0xC87F     [Cyclades Corporation PC300/TE (1 port)]
  0xF000-0xF00F     [Intel Corp. 82801BA IDE U100]

[admin@MikroTik] >
```

USB Port Information

Command name: ***/system resource usb print***

Description

Shows all USB ports available for the router.

Property Description

device (*read-only: text*) - number of device

name (*read-only: text*) - name of the USB port

speed (*read-only: integer*) - bandwidth speed at which the port works

vendor (*read-only: text*) - vendor name of the USB device

Example

To list all available USB ports:

```
[admin@MikroTik] system resource usb> print
 # DEVICE VENDOR                  NAME                    SPEED
 0 1:1                            USB OHCI Root Hub       12 Mbps
[admin@MikroTik] system resource usb>
```

PCI Information

Command name: ***/system resource pci print***

Property Description

device (*read-only: text*) - number of device

irq (*read-only: integer*) - IRQ number which this device uses

name (*read-only: text*) - name of the USB port

vendor (*read-only: text*) - vendor name of the USB device

Example

To see PCI slot details:

```
[admin@MikroTik] system resource pci> print
 # DEVICE   VENDOR                      NAME                         IRQ
 0 00:13.0  Compaq                      ZFMicro Chipset USB (rev... 12
 1 00:12.5  National Semi               SC1100 XBus (rev: 0)
 2 00:12.4  National Semi               SC1100 Video (rev: 1)
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
 3 00:12.3  National Semi               SCx200 Audio (rev: 0)
 4 00:12.2  National Semi               SCx200 IDE (rev: 1)
 5 00:12.1  National Semi               SC1100 SMI (rev: 0)
 6 00:12.0  National Semi               SC1100 Bridge (rev: 0)
 7 00:0e.0  Atheros Communications      AR5212 (rev: 1)            10
 8 00:0d.1  Texas Instruments           PCI1250 PC card Cardbus ... 11
 9 00:0d.0  Texas Instruments           PCI1250 PC card Cardbus ... 11
10 00:0c.0  National Semi               DP83815 (MacPhyter) Ethe... 10
11 00:0b.0  National Semi               DP83815 (MacPhyter) Ethe... 9
12 00:00.0  Cyrix Corporation           PCI Master (rev: 0)
[admin@MikroTik] system resource pci>
```

Reboot

Command name: ***/system reboot***

Description

The system reboot is required when upgrading or installing new software packages. The packages are installed during the system shutdown.

The reboot process sends termination signal to all running processes, unmounts the file systems, and reboots the router.

Notes

Only users, which are members of groups with reboot privileges are permitted to reboot the router.

Reboot can be called from scripts, in which case it does not prompt for confirmation.

Example

```
[admin@MikroTik] > system reboot
Reboot, yes? [y/N]: y
system will reboot shortly
[admin@MikroTik] >
```

Shutdown

Command name: ***/system shutdown***

Description

Before turning the power off for the router, the system should be brought to halt. The shutdown process sends termination signal to all running processes, unmounts the file systems, and halts the router.

For most systems, it is necessary to wait approximately 30 seconds for a safe power down.

Notes

Only users, which are members of groups with reboot privileges are permitted to shutdown the router.

Shutdown can be called from scripts, in which case it does not prompt for confirmation.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Example

```
[admin@MikroTik] > system shutdown
Shutdown, yes? [y/N]: y
system will shutdown promptly
[admin@MikroTik] >
```

Router Identity

Home menu level: ***/system identity***

Description

The router identity is displayed before the command prompt. It is also used for DHCP client as 'host name' parameter when reporting it to the DHCP server.

Example

To view the router identity:

```
[admin@MikroTik] > system identity print
    name: "MikroTik"
[admin@MikroTik] >
```

To set the router identity:

```
[admin@MikroTik] > system identity set name=Gateway
[admin@Gateway] >
```

Date and Time

Home menu level: ***/system clock***

Property Description

date (*text*) - date in format "mm/DD/YYY"

time (*time*) - time in format "HH:MM:SS"

time-zone (*text*) - UTC timezone in format "+HH:MM" or "-HH:MM"

Notes

It is recommended that you reboot the router after time change to obviate the possible errors in time measurments and logging.

Date and time settings become permanent and effect BIOS settings.

Example

To view the current date and time settings:

```
[admin@Local] system clock> print
         time: 08:26:37
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
         date: nov/18/2004
    time-zone: +00:00
[admin@Local] system clock>
```

To set the system date and time:

```
[admin@Local] system clock> set date=nov/22/2022 time=11:10:21 time-zone=+0
[admin@Local] system clock> print
         time: 11:10:25
         date: nov/22/2022
    time-zone: +00:00
[admin@Local] system clock>
```

Configuration Change History

Home menu level: Command name: ***/system history, /undo, /redo***

Description

The history of system configuration changes is held until the next router shutdown. The invoked commands can be 'undone' (in reverse order they have been invoked). The 'undone' commands may be 'redone' (in reverse order they have been 'undone').

Command Description

/redo - undoes previous '/undo' command

/system history print - print a list of last configuration changes, specifying whether the action can be undone or redone

/undo - undoes previous configuration changing command (except another '/undo' command)

Notes

Floating-undo actions are created within the current SAFE mode session. They are automatically converted to undoable and redoable when SAFE mode terminated successfully, and are all undone irreverively when SAFE mode terminated insuccessfully.

Undo command cannot undo commands past start of the SAFE mode.

Example

To show the list of configuration changes:

```
[admin@MikroTik] system history> print
Flags: U - undoable, R - redoable, F - floating-undo
   ACTION                                   BY                POLICY
 U system time zone changed                 admin             write
 U system time zone changed                 admin             write
 U system time zone changed                 admin             write
 U system identity changed                  admin             write
[admin@MikroTik] system clock>
```

What the **/undo** command does:

```
[admin@MikroTik] system history> print
Flags: U - undoable, R - redoable, F - floating-undo
   ACTION                                   BY                POLICY
 R system time zone changed                 admin             write
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  U system time zone changed                admin              write
  U system time zone changed                admin              write
  U system identity changed                 admin              write
[admin@MikroTik] system clock>
```

System Note

Home menu level: ***/system note***

Description

System note feature allows you to assign arbitrary text notes or messages that will be displayed on each login right after banner. For example, you may distribute warnings between system administrators this way, or describe what does that particular router actually do. To configure system note, you may upload a plain text file named **sys-note.txt** on the router's FTP server, or, additionally, edit the settings in this menu

Property Description

note (*text*; default: **""**) - the note

show-at-login (yes | no; default: **yes**) - whether to show system note on each login

Notes

If you want to enter or edit multiline system note, you may need to use embedded text editor: `/system note edit note`

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Bandwidth Test

Document revision 1.9 (Fri Nov 26 11:00:29 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
Server Configuration
  Property Description
  Notes
  Example
Client Configuration
  Property Description
  Example

General Information

Summary

The Bandwidth Tester can be used to monitor the throughput only to a remote MikroTik router (either wired or wireless) and thereby help to discover network "bottlenecks".

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/tool***
Standards and Technologies: ***TCP (RFC 793)***, ***UDP (RFC768)***
Hardware usage: *significant*

Related Documents

- ***Software Package Management***

Description

Protocol Description

The TCP test uses the standard TCP protocol with acknowledgments and follows the TCP algorithm on how many packets to send according to latency, dropped packets, and other features in the TCP algorithm. Please review the TCP protocol for details on its internal speed settings and how

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

to analyze its behavior. Statistics for throughput are calculated using the entire size of the TCP packet. As acknowledgments are an internal working of TCP, their size and usage of the link are not included in the throughput statistics. Therefore this statistic is not as reliable as the UDP statistic when estimating throughput.

The UDP tester sends 110% or more packets than currently reported as received on the other side of the link. To see the maximum throughput of a link, the packet size should be set for the maximum MTU allowed by the links which is usually 1500 bytes. There is no acknowledgment required by UDP; this implementation means that the closest approximation of the throughput can be seen.

Usage Notes

Caution! Bandwidth Test uses all available bandwidth (by default) and may impact network usability.

Bandwidth Test uses much resources. If you want to test real throughput of a router, you should run bandwidth test through it not from or to it. To do this you need at least 3 routers connected in chain: the Bandwidth Server, the given router and the Bandwidth Client:

Note that if you use UDP protocol then Bandwidth Test counts IP header+UDP header+UDP data. In case if you use TCP then Bandwidth Test counts only TCP data (TCP header and IP header are not included).

Server Configuration

Home menu level: ***/tool bandwidth-server***

Property Description

allocate-udp-ports-from - allocate UDP ports from

authenticate (*yes* | *no*; default: **yes**) - communicate only with authenticated (by valid username and password) clients

enable (*yes* | *no*; default: **no**) - enable client connections for bandwidth test

max-sessions - maximal number of bandwidth-test clients

Notes

The list of current connections can be obtained in **session** submenu

Example

Bandwidth Server:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] tool bandwidth-server> print
                    enabled: no
                authenticate: yes
    allocate-udp-ports-from: 2000
                max-sessions: 10
[admin@MikroTik] tool>
```

Active sessions:

```
[admin@MikroTik] tool> bandwidth-server session print
  # CLIENT          PROTOCOL DIRECTION USER
  0 35.35.35.1      udp      send      admin
  1 25.25.25.1      udp      send      admin
  2 36.36.36.1      udp      send      admin

[admin@MikroTik] tool>
```

To enable **bandwidth-test** server without client authentication:

```
[admin@MikroTik] tool bandwidth-server> set enabled=yes authenticate=no
[admin@MikroTik] tool bandwidth-server> print
                    enabled: yes
                authenticate: no
    allocate-udp-ports-from: 2000
                max-sessions: 10
[admin@MikroTik] tool>
```

Client Configuration

Command name: ***/tool bandwidth-test***

Property Description

address (*IP address*) - IP address of destination host

assume-lost-time (*time*; default: **0s**) - assume that connection is lost if Bandwidth Server is not responding for that time

direction (*receive*/*transmit*/*both*; default: **receive**) - the direction of the test

do (*name* | *string*; default: **""**) - script source

duration (*time*; default: **0s**) - duration of the test

- **0s** - test duration is not limited

interval (*time*: 20ms..5s; default: **1s**) - delay between reports (in seconds)

local-tx-speed (*integer*; default: **0**) - transfer test maximum speed (bits per second)

- **0** - no speed limitations

local-tx-size (*integer*: 40..64000) - local transmit packet size in bytes

password (*text*; default: **""**) - password for the remote user

protocol (*udp* | *tcp*; default: **udp**) - protocol to use

random-data (*yes* | *no*; default: **no**) - if random-data is set to yes, the payload of the bandwidth test packets will have incompressible random data so that links that use data compression will not distort the results (this is CPU intensive and random-data should be set to no for low speed CPUs)

remote-tx-speed (*integer*; default: **0**) - receive test maximum speed (bits per second)

- **0** - no speed limitations

remote-tx-size (*integer*: 40..64000) - remote transmit packet size in bytes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

user (*name*; default: **""**) - remote user

Example

To run 15-second long bandwidth-test to the **10.0.0.211** host sending and receiving **1000**-byte UDP packets and using username **admin** to connect

```
[admin@MikroTik] tool> bandwidth-test 10.0.0.211 duration=15s direction=both \
\... size=1000 protocol=udp user=admin
                status: done testing
                duration: 15s
              tx-current: 3.62Mbps
    tx-10-second-average: 3.87Mbps
        tx-total-average: 3.53Mbps
              rx-current: 3.33Mbps
    rx-10-second-average: 3.68Mbps
        rx-total-average: 3.49Mbps

[admin@MikroTik] tool>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

ICMP Bandwidth Test

Document revision 1.2 (Fri Mar 05 09:36:41 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
ICMP Bandwidth Test
  Description
  Property Description
  Example

General Information

Summary

The ICMP Bandwidth Tester (Ping Speed) can be used to approximately evaluate the throughput to **any** remote computer and thereby help to discover network 'bottlenecks'.

Specifications

Packages required: ***advanced-tools***
License required: ***level1***
Home menu level: ***/tool***
Standards and Technologies: ***ICMP (RFC792)***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***IP Addresses and ARP***
- ***Log Management***

ICMP Bandwidth Test

Description

The ICMP test uses two standard echo-requests per second. The time between these pings can be changed. Ping packet size variation makes it possible to approximately evaluate connection parameters and speed with different packet sizes. Statistics for throughput is calculated using the size of the ICMP packet, the interval between ICMP echo-request and echo-reply and the differences between parameters of the first and the second packet.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Property Description

do (*name*) - assigned name of the script to start

first-ping-size (*integer*: 32..64000; default: **32**) - first ICMP packet size

second-ping-size (*integer*: 32..64000; default: **1500**) - second ICMP packet size

time-between-pings (*integer*) - the time between the first and the second ICMP echo-requests in seconds. A new ICMP-packet pair will never be sent before the previous pair is completely sent and the algorithm itself will never send more than two requests in one second

once - specifies that the ping will be performed only once

interval (*time*: 20ms..5s) - time interval between two ping repetitions

Example

In the following example we will test the bandwidth to a host with IP address **159.148.60.2**. The interval between repetitions will be **1** second.

```
[admin@MikroTik] tool> ping-speed 159.148.60.2 interval=1s
    current: 2.23Mbps
    average: 2.61Mbps

[admin@MikroTik] tool>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Packet Sniffer

Document revision 1.5 (Thu May 20 14:56:46 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
Packet Sniffer Configuration
  Property Description
  Notes
  Example
Running Packet Sniffer
  Description
  Example
Sniffed Packets
  Description
  Property Description
  Example
Packet Sniffer Protocols
  Description
  Property Description
  Example
Packet Sniffer Host
  Description
  Property Description
  Example
Packet Sniffer Connections
  Description
  Property Description
  Example
  Sniff MAC Address

General Information

Summary

Packet sniffer is a feature that catches all the data travelling over the network, that it is able to get (when using switched network, a computer may catch only the data addressed to it or is forwarded through it).

Specifications

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Packages required: ***system***
License required: ***level1***
Home menu level: ***/tool sniffer***
Standards and Technologies: ***none***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***

Description

It allows you to "sniff" packets going through the router (and any other traffic that gets to the router, when there is no switching in the network) and view them using specific software.

Packet Sniffer Configuration

Home menu level: ***/tool sniffer***

Property Description

interface (*name* | *all*; default: **all**) - the name of the interface that receives the packets

only-headers (*yes* | *no*; default: **no**) - whether to save in the memory packets' headers only (not the whole packet)

memory-limit (*integer*; default: **10**) - maximum amount of memory to use. Sniffer will stop after this limit is reached

file-name (*text*; default: **""**) - the name of the file where the sniffed packets will be saved to

file-limit (*integer*; default: **10**) - the limit of the file in KB. Sniffer will stop after this limit is reached

streaming-enabled (*yes* | *no*; default: **no**) - whether to send sniffed packets to a remote server

streaming-server (*IP address*; default: **0.0.0.0**) - Tazmen Sniffer Protocol (TZSP) stream receiver

filter-stream (*yes* | *no*; default: **yes**) - whether to ignore sniffed packets that are destined to the stream server

filter-protocol (*all-frames* | *ip-only* | *mac-only-no-ip*; default: **ip-only**) - specific protocol group to filter

- **all-frames** - sniff all packets
- **ip-only** - sniff IP packets only
- **mac-only-no-ip** - sniff non-IP packets only

filter-address1 (*IP address/mask:port*; default: **0.0.0.0/0:0-65535**) - criterion of choosing the packets to process

filter-address2 (*IP address/mask:port*; default: **0.0.0.0/0:0-65535**) - criterion of choosing the packets to process

running (*yes* | *no*; default: **no**) - if the sniffer is started then the value is yes otherwise no

Notes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

filter-address1 and **filter-address2** are used to specify the two participients in communication (i.e. they will match only in the case if one of them matches the source address and the other one matches the destination address of a packet). These properties are taken in account only if **filter-protocol** is **ip-only**.

Not only **Ethernal** (***http://www.ethereal.com***) and **Packetyzer** (***http://www.packetyzer.com***) can receive the sniffer's stream but also MikroTik's program **trafr** (***http://www.mikrotik.com/download.html***) that runs on any IA32 Linux computer and saves received packets **libpcap** file format.

Example

In the following example **streaming-server** will be added, streaming will be enabled, **file-name** will be set to *test* and packet sniffer will be started and stopped after some time:

```
[admin@MikroTik] tool sniffer>set streaming-server=10.0.0.241 \
\... streaming-enabled=yes file-name=test
[admin@MikroTik] tool sniffer> prin
            interface: all
         only-headers: no
         memory-limit: 10
            file-name: "test"
           file-limit: 10
    streaming-enabled: yes
     streaming-server: 10.0.0.241
        filter-stream: yes
      filter-protocol: ip-only
      filter-address1: 0.0.0.0/0:0-65535
      filter-address2: 0.0.0.0/0:0-65535
              running: no
[admin@MikroTik] tool sniffer>start
[admin@MikroTik] tool sniffer>stop
```

Running Packet Sniffer

Command name: ***/tool sniffer start, /tool sniffer stop, /tool sniffer save***

Description

The commands are used to control runtime operation of the packet sniffer. The **start** command is used to start/reset sniffering, **stop** - stops sniffering. To save currently sniffed packets in a specific file **save** command is used.

Example

In the following example the packet sniffer will be started and after some time - stopped:

```
[admin@MikroTik] tool sniffer> start
[admin@MikroTik] tool sniffer> stop
```

Below the sniffed packets will be saved in the file named *test*:

```
[admin@MikroTik] tool sniffer> save file-name=test
[admin@MikroTik] tool sniffer> /file print
  # NAME                          TYPE         SIZE       CREATION-TIME
  0 test                          unknown      1350       apr/07/2003 16:01:52

[admin@MikroTik] tool sniffer>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Sniffed Packets

Home menu level: ***/tool sniffer packet***

Description

The submenu allows to see the list of sniffed packets.

Property Description

data (*read-only: text*) - specified data inclusion in packets

dst-address (*read-only: IP address*) - IP destination address

fragment-offset (*read-only: integer*) - IP fragment offset

identification (*read-only: integer*) - IP identification

ip-header-size (*read-only: integer*) - the size of IP header

ip-packet-size (*read-only: integer*) - the size of IP packet

ip-protocol (*ip* | *icmp* | *igmp* | *ggp* | *ipencap* | *st* | *tcp* | *egp* | *pup* | *udp* | *hmp* | *xns-idp* | *rdp* | *iso-tp4* | *xtp* | *ddp* | *idrp-cmtp* | *gre* | *esp* | *ah* | *rspf* | *vmtp* | *ospf* | *ipip* | *encap*) - the name/number of IP protocol

- **ip** - Internet Protocol
- **icmp** - Internet Control Message Protocol
- **igmp** - Internet Group Management Protocol
- **ggp** - Gateway-Gateway Protocol
- **ipencap** - IP Encapsulated in IP
- **st** - st datagram mode
- **tcp** - Transmission Control Protocol
- **egp** - Exterior Gateway Protocol
- **pup** - Parc Universal packet Protocol
- **udp** - User Datagram Protocol
- **hmp** - Host Monitoring Protocol
- **xns-idp** - Xerox ns idp
- **rdp** - Reliable Datagram Protocol
- **iso-tp4** - ISO Transport Protocol class 4
- **xtp** - Xpress Transfer Protocol
- **ddp** - Datagram Delivery Protocol
- **idpr-cmtp** - idpr Control Message Transport
- **gre** - General Routing Encapsulation
- **esp** - IPsec ESP protocol
- **ah** - IPsec AH protocol
- **rspf** - Radio Shortest Path First
- **vmtp** - Versatile Message Transport Protocol

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **ospf** - Open Shortest Path First
- **ipip** - IP encapsulation (protocol 4)
- **encap** - IP encapsulation (protocol 98)

protocol (*read-only: ip | arp | rarp | ipx | ipv6*) - the name/number of ethernet protocol
- **ip** - Internet Protocol
- **arp** - Address Resolution Protocol
- **rarp** - Reverse Address Resolution Protocol
- **ipx** - Internet Packet exchange protocol
- **ipv6** - Internet Protocol next generation

size (*read-only: integer*) - size of packet

src-address (*IP address*) - source address

time (*read-only: time*) - time when packet arrived

tos (*read-only: integer*) - IP Type Of Service

ttl (*read-only: integer*) - IP Time To Live

Example

In the example below it's seen, how to get the list of sniffed packets:

```
[admin@MikroTik] tool sniffer packet> pr
  # TIME    INTERFACE SRC-ADDRESS            DST-ADDRESS            IP-.. SIZE
  0 0.12    ether1    10.0.0.241:1839        10.0.0.181:23 (telnet) tcp   46
  1 0.12    ether1    10.0.0.241:1839        10.0.0.181:23 (telnet) tcp   40
  2 0.12    ether1    10.0.0.181:23 (telnet) 10.0.0.241:1839        tcp   78
  3 0.292   ether1    10.0.0.181             10.0.0.4               gre   88
  4 0.32    ether1    10.0.0.241:1839        10.0.0.181:23 (telnet) tcp   40
  5 0.744   ether1    10.0.0.144:2265        10.0.0.181:22 (ssh)    tcp   76
  6 0.744   ether1    10.0.0.144:2265        10.0.0.181:22 (ssh)    tcp   76
  7 0.744   ether1    10.0.0.181:22 (ssh)    10.0.0.144:2265        tcp   40
  8 0.744   ether1    10.0.0.181:22 (ssh)    10.0.0.144:2265        tcp   76
-- more
```

Packet Sniffer Protocols

Home menu level: ***/tool sniffer protocol***

Description

In this submenu you can see all kind of protocols that have been sniffed.

Property Description

bytes (*integer*) - total number of data bytes

protocol (*read-only: ip | arp | rarp | ipx | ipv6*) - the name/number of ethernet protocol
- **ip** - Internet Protocol
- **arp** - Address Resolution Protocol
- **rarp** - Reverse Address Resolution Protocol
- **ipx** - Internet Packet exchange protocol

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **ipv6** - Internet Protocol next generation

ip-protocol (*ip* | *icmp* | *igmp* | *ggp* | *ipencap* | *st* | *tcp* | *egp* | *pup* | *udp* | *hmp* | *xns-idp* | *rdp* | *iso-tp4* | *xtp* | *ddp* | *idrp-cmtp* | *gre* | *esp* | *ah* | *rspf* | *vmtp* | *ospf* | *ipip* | *encap*) - the name/number of IP protocol

- **ip** - Internet Protocol
- **icmp** - Internet Control Message Protocol
- **igmp** - Internet Group Management Protocol
- **ggp** - Gateway-Gateway Protocol
- **ipencap** - IP Encapsulated in IP
- **st** - st datagram mode
- **tcp** - Transmission Control Protocol
- **egp** - Exterior Gateway Protocol
- **pup** - Parc Universal packet Protocol
- **udp** - User Datagram Protocol
- **hmp** - Host Monitoring Protocol
- **xns-idp** - Xerox ns idp
- **rdp** - Reliable Datagram Protocol
- **iso-tp4** - ISO Transport Protocol class 4
- **xtp** - Xpress Transfer Protocol
- **ddp** - Datagram Delivery Protocol
- **idpr-cmtp** - idpr Control Message Transport
- **gre** - General Routing Encapsulation
- **esp** - IPsec ESP protocol
- **ah** - IPsec AH protocol
- **rspf** - Radio Shortest Path First
- **vmtp** - Versatile Message Transport Protocol
- **ospf** - Open Shortest Path First
- **ipip** - IP encapsulation
- **encap** - IP encapsulation

packets (*integer*) - the number of packets

port (*name*) - the port of TCP/UDP protocol

share (*integer*) - specific type of traffic compared to all traffic in bytes

Example

```
[admin@MikroTik] tool sniffer protocol> print
  # PROTOCOL IP-PR... PORT          PACKETS   BYTES   SHARE
  0 ip                              77        4592    100 %
  1 ip       tcp                    74        4328    94.25 %
  2 ip       gre                    3         264     5.74 %
  3 ip       tcp      22 (ssh)      49        3220    70.12 %
  4 ip       tcp      23 (telnet)   25        1108    24.12 %

[admin@MikroTik] tool sniffer protocol>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Packet Sniffer Host

Home menu level: ***/tool sniffer host***

Description

The submenu shows the list of hosts that were participating in data excange you've sniffed.

Property Description

address (*read-only: IP address*) - IP address of the host
peek-rate (*read-only: integer/integer*) - the maximum data-rate received/transmitted
rate (*read-only: integer*/*integer*) - current data-rate received/transmitted
total (*read-only: integer*/*integer*) - total packets received/transmitted

Example

In the following example we'll see the list of hosts:

```
[admin@MikroTik] tool sniffer host> print
  # ADDRESS       RATE          PEEK-RATE           TOTAL
  0 10.0.0.4      0bps/0bps     704bps/0bps         264/0
  1 10.0.0.144    0bps/0bps     6.24kbps/12.2kbps   1092/2128
  2 10.0.0.181    0bps/0bps     12.2kbps/6.24kbps   2994/1598
  3 10.0.0.241    0bps/0bps     1.31kbps/4.85kbps   242/866

[admin@MikroTik] tool sniffer host>
```

Packet Sniffer Connections

Home menu level: ***/tool sniffer connection***

Description

Here you can get a list of the connections that have been watched during the sniffing time.

Property Description

active (*read-only: yes* | *no*) - if yes the find active connections
bytes (*read-only: integer*) - bytes in the current connection
dst-address (*read-only: IP address*) - destination address
mss (*read-only: integer*) - Maximum Segment Size
resends (*read-only: integer*) - the number of packets resends in the current connection
src-address (*read-only: IP address*) - source address

Example

The example shows how to get the list of connections:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] tool sniffer connection> print
Flags: A - active
  #   SRC-ADDRESS       DST-ADDRESS             BYTES     RESENDS   MSS
  0 A 10.0.0.241:1839   10.0.0.181:23 (telnet)  6/42      60/0      0/0
  1 A 10.0.0.144:2265   10.0.0.181:22 (ssh)     504/252   504/0     0/0

[admin@MikroTik] tool sniffer connection>
```

Sniff MAC Address

You can also see the source and destination MAC Addresses. To do so, at first stop the sniffer if it is running, and select a specific interface:

```
[admin@MikroTik] tool sniffer> stop
[admin@MikroTik] tool sniffer> set interface=bridge1
[admin@MikroTik] tool sniffer> start
[admin@MikroTik] tool sniffer> print
            interface: bridge1
         only-headers: no
         memory-limit: 10
            file-name:
           file-limit: 10
    streaming-enabled: no
     streaming-server: 0.0.0.0
        filter-stream: yes
      filter-protocol: ip-only
      filter-address1: 0.0.0.0/0:0-65535
      filter-address2: 0.0.0.0/0:0-65535
              running: yes
[admin@MikroTik] tool sniffer>
```

Now you have the source and destination MAC Addresses:

```
[admin@MikroTik] tool sniffer packet> print detail
 0 time=0 src-mac-address=00:0C:42:03:02:C7 dst-mac-address=00:30:4F:08:3A:E7
   interface=bridge1 src-address=10.5.8.104:1125
   dst-address=10.1.0.172:3987 (winbox-tls) protocol=ip ip-protocol=tcp
   size=146 ip-packet-size=146 ip-header-size=20 tos=0 identification=5088
   fragment-offset=0 ttl=126

 1 time=0 src-mac-address=00:30:4F:08:3A:E7 dst-mac-address=00:0C:42:03:02:C7
   interface=bridge1 src-address=10.1.0.172:3987 (winbox-tls)
   dst-address=10.5.8.104:1125 protocol=ip ip-protocol=tcp size=253
   ip-packet-size=253 ip-header-size=20 tos=0 identification=41744
   fragment-offset=0 ttl=64

 2 time=0.071 src-mac-address=00:0C:42:03:02:C7
   dst-mac-address=00:30:4F:08:3A:E7 interface=bridge1
   src-address=10.5.8.104:1125 dst-address=10.1.0.172:3987 (winbox-tls)
   protocol=ip ip-protocol=tcp size=40 ip-packet-size=40 ip-header-size=20
   tos=0 identification=5089 fragment-offset=0 ttl=126

 3 time=0.071 src-mac-address=00:30:4F:08:3A:E7
   dst-mac-address=00:0C:42:03:02:C7 interface=bridge1
   src-address=10.1.0.172:3987 (winbox-tls) dst-address=10.5.8.104:1125
   protocol=ip ip-protocol=tcp size=213 ip-packet-size=213 ip-header-size=20
   tos=0 identification=41745 fragment-offset=0 ttl=64

-- [Q quit|D dump|down]
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Ping

Document revision 1 (Mon Jul 19 09:36:24 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
The Ping Command
  Property Description
  Notes
  Example of ping command
  Resolve IP address:
  'Ping', using arp requests:
MAC Ping Server
  Property Description
  Example

General Information

Summary

Ping uses Internet Control Message Protocol (ICMP) Echo messages to determine if a remote host is active or inactive and to determine the round-trip delay when communicating with it.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/, /tool mac-server ping***
Standards and Technologies: ***ICMP***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***

Description

Ping sends ICMP echo (ICMP type 8) message to the host and waits for the ICMP echo-reply (ICMP type 0) from that host. The interval between these events is called round trip. If the response (that is called pong) has not come until the end of the interval, we assume it has timed out. The second significant parameter reported is ttl (Time to Live). Is is decremented at each machine in

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

which the packet is processed. The packet will reach its destination only when the ttl is greater than the number of routers between the source and the destination.

The Ping Command

Command name: ***/ping***

Property Description

arp-interface (*name*) - ping, using ARP requests on this interface, instead of ICMP requests.

(*IP address* | *MAC address*) - IP or MAC address for destination host

count (*integer*; default: **0**) - how many times ICMP packets will be sent

- **0** - Ping continues till [Ctrl]+[C] is pressed

do-not-fragment - if added, packets will not be fragmented

interval (*time*: 10ms..5s; default: **1s**) - delay between messages

size (*integer*: 28..65535; default: **64**) - size of the IP packet (in bytes, including the IP and ICMP headers)

ttl (*integer*: 1..255; default: **255**) - time To Live (TTL) value of the ICMP packet

src-address (*IP address*) - Source address for ping

Notes

If DNS service is configured, it is possible to ping by DNS address. To do it from **Winbox**, you should resolve DNS address first, pressing right mouse button over its address and choosing **Lookup Address**.

You cannot ping with packets larger that the MTU of that interface, so the packet **size** should always be equal or less than MTU. If 'pinging' by MAC address, minimal packet size iz 50 bytes.

Only neighbour MikroTik RouterOS routers with MAC-ping feature enabled can be 'pinged' by MAC address.

Example of ping command

An example of Ping command:

```
/pi 159.148.95.16 count=5 interval=500ms
159.148.95.16 64 byte ping: ttl=59 time=21 ms
159.148.95.16 ping timeout
159.148.95.16 ping timeout
159.148.95.16 ping timeout
159.148.95.16 64 byte ping: ttl=59 time=16 ms
5 packets transmitted, 2 packets received, 60% packet loss
round-trip min/avg/max = 16/18.5/21 ms
[admin@MikroTik] >
```

Resolve IP address:

To resolve IP address from a DNS name, type the command:

```
/ping www.google.lv
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

and press the [Tab] key:

```
[admin@MikroTik] > /ping 66.102.11.104
```

The DNS name **www.google.lv** changed to IP address 66.102.11.104!

'Ping', using arp requests:

To ping a host in our local network, using ARP requests instead of ICMP:

```
/ping 10.5.8.130 arp-interface=local
10.5.8.130 with hw-addr 00:30:4F:14:AB:58 ping time=1 ms
10.5.8.130 with hw-addr 00:30:4F:14:AB:58 ping time=1 ms
10.5.8.130 with hw-addr 00:30:4F:14:AB:58 ping time=1 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 1/1.0/1 ms
[admin@MikroTik] >
```

MAC Ping Server

Home menu level: ***/tool mac-server ping***

Property Description

enabled (yes | no; default: **yes**) - whether MAC pings to this router are allowed

Example

To disable MAC pings:

```
[admin@MikroTik] tool mac-server ping> set enabled=no
[admin@MikroTik] tool mac-server ping> print
    enabled: no
[admin@MikroTik] tool mac-server ping>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Torch (Realtime Traffic Monitor)

Document revision 1.8 (Fri Nov 05 12:25:04 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
The Torch Command
  Property Description
  Notes
  Example

General Information

Summary

Realtime traffic monitor may be used to monitor the traffic flow through an interface.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/tool***
Standards and Technologies: ***none***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***

Description

Realtime Traffic Monitor called also torch is used for monitoring traffic that is going through an interface. You can monitor traffic classified by protocol name, source address, destination address, port. Torch shows the protocols you have chosen and mean transmitted and received data rate for each of them.

The Torch Command

Command name: ***/tool torch***

Property Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

interface (*name*) - the name of the interface to monitor

protocol (*any* | *any-ip* | *ddp* | *egp* | *encap* | *ggp* | *gre* | *hmp* | *icmp* | *idpr-cmtp* | *igmp* | *ipencap* | *ipip* | *ipsec-ah* | *ipsec-esp* | *iso-tp4* | *ospf* | *pup* | *rdp* | *rspf* | *st* | *tcp* | *udp* | *vmtp* | *xns-idp* | *xtp*) - the name or number of the protocol

- **any** - any ethernet or IP protocol
- **any-ip** - any IP protocol

port (*name* | *integer*) - the name or number of the port

src-address (*IP address/mask*) - source address and network mask to filter the traffic only with such an address, any source address: 0.0.0.0/0

dst-address (*IP address/mask*) - destination address and network mask to filter the traffic only with such an address, any destination address: 0.0.0.0/0

average-seconds (*integer*: 1..10) - the average speed will be shown in the last average seconds

freeze-frame-interval (*time*) - time in seconds for which the screen output is paused

Notes

If there will be specific port given, then only **tcp** and **udp** protocols will be filtered, i.e., the name of the **protocol** can be **any**, **any-ip**, **tcp**, **udp**.

Except TX and RX, there will be only the field you've specified in command line in the command's output (e.g., you will get **PROTOCOL** column only in case if **protocol** property is explicitly specified).

Example

The following example monitors the traffic that goes through the **ether1** interface generated by **telnet** protocol:

```
[admin@MikroTik] tool> torch ether1 port=telnet
 SRC-PORT                    DST-PORT                    TX         RX
 1439                        23 (telnet)                 1.7kbps    368bps

[admin@MikroTik] tool>
```

To see what IP protocols are going through the **ether1** interface:

```
[admin@MikroTik] tool> torch ether1 protocol=any-ip
 PRO.. TX         RX
 tcp   1.06kbps   608bps
 udp   896bps     3.7kbps
 icmp  480bps     480bps
 ospf  0bps       192bps

[admin@MikroTik] tool>
```

To see what IP protocols are interacting with **10.0.0.144/32** host connected to the **ether1** interface:

```
[admin@MikroTik] tool> torch ether1 src-address=10.0.0.144/32 protocol=any
 PRO.. SRC-ADDRESS     TX         RX
 tcp   10.0.0.144      1.01kbps   608bps
 icmp  10.0.0.144      480bps     480bps

[admin@MikroTik] tool>
```

To see what tcp/udp protocols are going through the **ether1** interface:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] tool> torch ether1 protocol=any-ip port=any
 PRO.. SRC-PORT                  DST-PORT                  TX         RX
 tcp   3430                      22 (ssh)                  1.06kbps   608bps
 udp   2812                      1813 (radius-acct)        512bps     2.11kbps
 tcp   1059                      139 (netbios-ssn)         248bps     360bps
[admin@MikroTik] tool>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Traceroute

Document revision 1.8 (Fri Nov 26 13:00:20 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
The Traceroute Command
  Property Description
  Notes
  Example

General Information

Summary

Traceroute determines how packets are being routed to a particular host.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/tool***
Standards and Technologies: ***ICMP***, ***UDP***, ***Traceroute***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***IP Addresses and ARP***
- ***Firewall Filters***
- ***Ping***

Description

Traceroute is a TCP/IP protocol-based utility, which allows user to determine how packets are being routed to a particular host. Traceroute works by increasing the time-to-live value of packets and seeing how far they get until they reach the given destination; thus, a lengthening trail of hosts passed through is built up.

Traceroute shows the number of hops to the given host address of every passed gateway. Traceroute

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

utility sends packets three times to each passed gateway so it shows three timeout values for each gateway in ms.

The Traceroute Command

Command name: ***/tool traceroute***

Property Description

(*IP address*) - IP address of the host you are tracing route to

port (*integer*: 0..65535) - UDP port number

protocol (*UDP* | *ICMP*) - type of protocol to use. If one fails (for example, it is blocked by a firewall), try the other

size (*integer*: 28..1500; default: **64**) - packet size in bytes

timeout (*time*: 1s..8s; default: **1s**) - response waiting timeout, i.e. delay between messages

tos (*integer*: 0..255; default: **0**) - Type Of Service - parameter of IP packet

use-dns (*yes* | *no*; default: **no**) - specifies whether to use DNS server, which can be set in /ip dns menu

src-address (*IP address*) - change the source address of the packet

max-hops (*integer*) - utmost hops through which packet can be reached

Notes

Traceroute session may be stopped by pressing [Ctrl]+[C].

Example

To trace the route to 216.239.39.101 host using ICMP protocol with packet size of 64 bytes, setting ToS field to 8 and extending the timeout to 4 seconds:

```
[admin@MikroTik] tool> traceroute 216.239.39.101 protocol=icmp size=64 tos=8 timeout=4s
     ADDRESS                                  STATUS
   1 159.148.60.227   3ms    3ms     3ms
   2 195.13.173.221  80ms  169ms    14ms
   3 195.13.173.28    6ms    4ms     4ms
   4 195.158.240.21 111ms  110ms   110ms
   5 213.174.71.49 124ms  120ms   129ms
   6 213.174.71.134 139ms  146ms   135ms
   7 213.174.70.245 132ms  131ms   136ms
   8 213.174.70.58 211ms  215ms   215ms
   9 195.158.229.130 225ms  239ms      0s
  10 216.32.223.114 283ms  269ms   281ms
  11 216.32.132.14 267ms  260ms   266ms
  12 209.185.9.102 296ms  296ms   290ms
  13 216.109.66.1 288ms  297ms   294ms
  14 216.109.66.90 297ms  317ms   319ms
  15 216.239.47.66 137ms  136ms   134ms
  16 216.239.47.46 135ms  134ms   134ms
  17 216.239.39.101 134ms  134ms   135ms
[admin@MikroTik] tool>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Scheduler

Document revision 0.9 (Wed Nov 24 12:48:55 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
Network Watching Tool
  Specifications
  Description
  Property Description
  Example

General Information

Summary

System Scheduler executes scripts at designated time.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/system scheduler***
Standards and Technologies: ***None***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Scripting Examples***
- ***Scripting Examples***

Network Watching Tool

Specifications

Packages required: ***advanced-tools***
License required: ***level1***
Home menu level: ***/tool netwatch***
Standards and Technologies: ***none***
Hardware usage: ***Not significant***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

Netwatch monitors state of hosts on the network. It does so by sending ICMP pings to the list of specified IP addresses. For each entry in netwatch table you can specify IP address, ping interval and console scripts. The main advantage of netwatch is it's ability to issue arbitrary console commands on host state changes.

Property Description

down-script (*name*) - a console script that is executed once when state of a host changes from unknown or up to down

host (*IP address*; default: **0.0.0.0**) - IP address of host that should be monitored

interval (*time*; default: **1s**) - the time between pings. Lowering this will make state changes more responsive, but can create unnecessary traffic and consume system resources

since (*read-only: time*) - indicates when state of the host changed last time

status (*read-only: up* | *down* | *unknown*) - shows the current status of the host

- **up** - the host is up
- **down** - the host is down
- **unknown** - after any properties of this list entry were changed, or the item is enabled or disabled

timeout (*time*; default: **1s**) - timeout for each ping. If no reply from a host is received during this time, the host is considered unreachable (down)

up-script (*name*) - a console script that is executed once when state of a host changes from unknown or down to up

Example

This example will run the scripts gw_1 or gw_2 which change the default gateway depending on the status of one of the gateways:

```
[admin@MikroTik] system script> add name=gw_1 source={/ip route set
{... [/ip route find dst 0.0.0.0] gateway 10.0.0.1}
[admin@MikroTik] system script> add name=gw_2 source={/ip route set
{.. [/ip route find dst 0.0.0.0] gateway 10.0.0.217}
[admin@MikroTik] system script> /tool netwatch
[admin@MikroTik] tool netwatch> add host=10.0.0.217 interval=10s timeout=998ms \
\... up-script=gw_2 down-script=gw_1
[admin@MikroTik] tool netwatch> print
Flags: X - disabled
  #   HOST       TIMEOUT            INTERVAL         STATUS
  0   10.0.0.217      997ms            10s                up
[admin@MikroTik] tool netwatch> print detail
Flags: X - disabled
  0   host=10.0.0.217 timeout=997ms interval=10s since=feb/27/2003 14:01:03
      status=up up-script=gw_2 down-script=gw_1

[admin@MikroTik] tool netwatch>
```

Without scripts, netwatch can be used just as an information tool to see which links are up, or which specific hosts are running at the moment.

Let's look at the example above - it changes default route if gateway becomes unreachable. How it's

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

done? There are two scripts. The script "gw_2" is executed once when status of host changes to **up**. In our case, it's equivalent to entering this console command:

```
[admin@MikroTik] > /ip route set [/ip route find dst 0.0.0.0] gateway 10.0.0.217
```

The **/ip route find dst 0.0.0.0** command returns list of all routes whose **dst-address** value is **0.0.0.0**. Usually, that is the default route. It is substituted as first argument to **/ip route set** command, which changes gateway of this route to 10.0.0.217

The script "gw_1" is executed once when status of host becomes **down**. It does the following:

```
[admin@MikroTik] > /ip route set [/ip route find dst 0.0.0.0] gateway 10.0.0.1
```

It changes the default gateway if 10.0.0.217 address has become unreachable.

Here is another example, that sends e-mail notification whenever the 10.0.0.215 host goes down:

```
[admin@MikroTik] system script> add name=e-down source={/tool e-mail send
{... from="rieks@mt.lv" server="159.148.147.198" body="Router down"
{... subject="Router at second floor is down" to="rieks@latnet.lv"}
[admin@MikroTik] system script> add name=e-up source={/tool e-mail send
{... from="rieks@mt.lv" server="159.148.147.198" body="Router up"
{.. subject="Router at second floor is up" to="rieks@latnet.lv"}
[admin@MikroTik] system script>
[admin@MikroTik] system script> /tool netwatch
[admin@MikroTik] system netwatch> add host=10.0.0.215 timeout=999ms \
\... interval=20s up-script=e-up down-script=e-down
[admin@MikroTik] tool netwatch> print detail
Flags: X - disabled
  0   host=10.0.0.215 timeout=998ms interval=20s since=feb/27/2003 14:15:36
      status=up up-script=e-up down-script=e-down

[admin@MikroTik] tool netwatch>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Serial Port Monitor

Document revision 1 (Mon Jul 11 10:17:08 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
Sigwatch
  Description
  Property Description
  Notes
  Example

General Information

Summary

Serial port monitoring utility monitors state of attached asynchronous serial ports and generates system events upon state change.

Specifications

Packages required: ***advanced-tools***
License required: ***level1***
Home menu level: ***/tool sigwatch***
Standards and Technologies: ***none***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***Scripting Host***

Sigwatch

Description

Sigwatch monitors state of the serial port pins.

Property Description

count (*read-only: integer*) - how many times the event for this item was triggered. Count is reset on

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

reboot and on most item configuration changes

log (*yes* | *no*; default: **no**) - whether to add a message in form of name-of-sigwatch-item: signal changed [to high | to low] to System-Info facility whenever this sigwatch item is triggered

name (*name*) - name of the sigwatch item

on-condition (*on* | *off* | *change*; default: **on**) - on what condition to trigger action of this item

- **on** - trigger when state of pin changes to high
- **off** - trigger when state of pin changes to low
- **change** - trigger whenever state of pin changes. If state of pin changes rapidly, there might be triggered only one action for several state changes

port (*name*) - serial port name to monitor

script (*name*) - script to execute when this item is trigered

signal (*dtr* | *rts* | *cts* | *dcd* | *ri* | *dsr*; default: **rts**) - name of signal of number of pin (for standard 9-pin connector) to monitor

- **dtr** - Data Terminal Ready (pin #4)
- **rts** - Request To Send (pin #7)
- **cts** - Clear To Send (pin #8)
- **dcd** - Data Carrier Detect (pin #1)
- **ri** - Ring Indicator (pin #9)
- **dsr** - Data Set Ready (pin #6)

state (*read-only: text*) - last remembered state of monitored signal

Notes

You can type actual script source instead of the script name from **/system script** list.

Example

In the following example we will add a new sigwatch item that monitors whether the port **serial1** has cts signal.

```
[admin@10.179] tool sigwatch> pr
Flags: X - disabled
  #   NAME                                  PORT    SIGNAL     ON-CONDITION LOG
  0   test                                  serial1 cts        change       no
[admin@MikroTik] tool sigwatch>
```

By typing a command **print detail interval=1s**, we can check whether a cable is connected or it is not. See the **state** argument - if the cable is connected to the serial port, it shows **on**, otherwise it will be **off**.

```
[admin@MikroTik] tool sigwatch> print detail
Flags: X - disabled
  0   name="test" port=serial1 signal=cts on-condition=change log=no script=""
      count=1 state=on

[admin@MikroTik] tool sigwatch> print detail
Flags: X - disabled
  0   name="test" port=serial1 signal=cts on-condition=change log=no script=""
      count=1 state=on

[admin@MikroTik] tool sigwatch> print detail
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
Flags: X - disabled
  0   name="test" port=serial1 signal=cts on-condition=change log=no script=""
      count=2 state=off

[admin@MikroTik] tool sigwatch> print detail
Flags: X - disabled
  0   name="test" port=serial1 signal=cts on-condition=change log=no script=""
      count=2 state=off

[admin@MikroTik] tool sigwatch>
```

In the **port** menu you can see what **signal** is used by serial cable. For example, without any cables it looks like this:

```
[admin@MikroTik] port> print stats
  0 name="serial0" line-state=dtr,rts

  1 name="serial1" line-state=dtr,rts
[admin@MikroTik] port>
```

But after adding a serial cable to the serial port:

```
[admin@MikroTik] port> print stats
  0 name="serial0" line-state=dtr,rts

  1 name="serial1" line-state=dtr,rts,cts
[admin@MikroTik] port>
```

This means that the line-state besides the **dtr** and **rts** signals has also **cts** when a serial cable is connected.

The example below will execute a script whenever **on-condition** changes to **off**:

```
[admin@10.MikroTik] tool sigwatch> pr detail
Flags: X - disabled
  0   name="cts_rest" port=serial1 signal=cts on-condition=off log=no
      script=/system shutdown count=0 state=on
[admin@10.MikroTik] tool sigwatch>
```

It means that if a serial cable is connected to the serial port, all works fine, but as soon as it is disconnected, the router shuts down. It will continue all the time until the serial cable will not be connected again.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Scripting Host

Document revision 2.7 (Thu Sep 22 13:33:55 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
Console Command Syntax
  Description
  Notes
  Example
Expression Grouping
  Description
  Notes
  Example
Variables
  Description
  Notes
  Example
Command Substitution and Return Values
  Description
  Example
Operators
  Description
  Command Description
  Notes
  Example
Data types
  Description
Command Reference
  Description
  Command Description
Special Commands
  Description
  Notes
  Example
Additional Features
  Description
Script Repository
  Description
  Property Description
  Command Description
  Notes
  Example

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Task Management
  Description
  Property Description
  Example
Script Editor
  Description
  Command Description
  Notes
  Example

General Information

Summary

This manual provides introduction to RouterOS built-in powerful scripting language.

Scripting host provides a way to automate some router maintenance tasks by means of executing user-defined scripts bounded to some event occurence. A script consists of configuration commands and expressions (ICE - internal console expression). The configuration commands are standard RouterOS commands, e.g. `/ip firewall filter add chain=forward protocol=gre action=drop` that are described in the relevant manuals, while expressions are prefixed with **:** and are accessible from all submenus.

The events used to trigger script execution include, but are not limited to the System Scheduler, the Traffic Monitoring Tool, and the Netwatch Tool generated events.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/system script***
Standards and Technologies: ***None***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***System Scheduler***
- ***Network Monitor***
- ***Traffic Monitor***
- ***Serial Port Monitor***

Console Command Syntax

Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Console commands are made of the following parts, listed in the order you type them in console:

- **prefix** - indicates whether the command is an ICE, like **:** in `:put` or that the command path starts from the root menu level, like **/** in

```
[admin@MikroTik] ip firewall mangle> /ping 10.0.0.1
```

- **path** - a relative path to the desired menu level, like **.. filter** in

```
[admin@MikroTik] ip firewall mangle> .. filter print
```

- **path_args** - this part is required to select some menu levels, where the actual path can vary across different user inputs, like **mylist** in

```
[admin@MikroTik] ip firewall mangle> /routeing prefix-list list mylist
```

- **action** - one of the actions available at the specified menu level, like **add** in

```
[admin@MikroTik] ip firewall mangle> /ip firewall filter add chain=forward action=drop
```

- **unnamed parameter** - these are required by some actions and should be entered in fixed order after the action name, like in **10.0.0.1** in

```
[admin@MikroTik] ip firewall mangle> /ping 10.0.0.1
```

- **name[=value]** - a sequence of parameter names followed by respective values, if required, like **ssid=myssid** in

```
/interface wireless set wlan1 ssid=myssid
```

Notes

Variable substitution, command substitution and expressions are allowed only for **path_args** and **unnamed parameter** values. **prefix**, **path**, **action** and **name[=value]** pairs can be given only directly, as a word. Therefore, `:put (1 + 2)` is valid and `:("pu" . "t") 3` is not.

Example

The parts of internal console commands are futher explained in the following examples:

```
/ping 10.0.0.1 count=5
```

prefix	/
action	ping
unnamed parameter	10.0.0.1
name[=value]	count=5

```
.. ip firewall rule input
```

path	.. ip firewall rule
path_args	input

```
:for i from=1 to=10 do={:put $i}
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

prefix	:
action	for
unnamed parameter	i
pname[=value]	from=1 to=10 do={:put $i}

```
/interface monitor-traffic ether1,ether2,ipip1
```

prefix	/
path	interface
action	monitor-traffic
unnamed parameter	ether1,ether2,ipip1

Expression Grouping

Description

This feature provides an easy way to execute commands from within one command level, by enclosing them in braces '{ }'.

Notes

Subsequent script commands are executed from the same menu level as the entire script. Consider the following example:

```
[admin@MikroTik] ip route> /user {
{... /ip route
{... print}
Flags: X - disabled
 #   NAME                                                    GROUP ADDRESS
 0   ;;; system default user
     admin                                                   full  0.0.0.0/0
 1   uuu                                                     full  0.0.0.0/0
[admin@MikroTik] ip route>
```

Although the current command level is changed to **/ip route**, it has no effect on next commands entered from prompt, therefore **print** command is still considered to be **/user print**.

Example

The example below demonstrates how to add two users to the **user** menu.

```
[admin@MikroTik] ip route> /user {
{... add name=x password=y group=write
{... add name=y password=z group=read
{... print}
Flags: X - disabled
 #   NAME                                                    GROUP ADDRESS
 0   ;;; system default user
     admin                                                   full  0.0.0.0/0
 1   x                                                       write 0.0.0.0/0
 2   y                                                       read  0.0.0.0/0
[admin@MikroTik] ip route>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Variables

Description

RouterOS scripting language suports two types of variables, which are global (system wide) and local (accessible only within the current script), respectively. A variable can be referenced by '$' (dollar) sign followed by the name of the variable with the exception of **set** and **unset** commands that take variable name without preceding dollar sign. Variable names should be composed of contain letters, digits and '-' character. A variable must be declared prior to using it in scripts. There are four types of declaration available:

- **global** - defined by global keyword, global variables can be accessed by all scripts and console logins on the same router. However, global variables are not kept across reboots.
- **local** - defined by local keyword, local variables are not shared with any other script, other instance of the same script or other console logins. The value of local variable value is lost when script finishes.
- **loop index variables** - defined within for and foreach statements, these variables are used only in do block of commands and are removed after command completes.
- **monitor variables** - some monitor commands that have do part can also introduce variables. You can obtain a list of available variables by placing :environment print statement inside the do block of commands.

You can assign a new value to variable using **set** action. It takes two unnamed parameters: the name of the variable and the new value of the variable. If a variable is no longer needed, it's name can be freed by **:unset** command. If you free local variable, it's value is lost. If you free global variable, it's value is still kept in router, it just becomes inaccessible from current script.

Notes

Loop variables "shadows" already introduced variables with the same name.

Example

```
[admin@MikroTik] ip route> /
[admin@MikroTik] > :global g1 "this is global variable"
[admin@MikroTik] > :put $g1
this is global variable
[admin@MikroTik] >
```

Command Substitution and Return Values

Description

Some console commands are most useful if their output can be feed to other commands as an argument value. In RouterOS console this is done by using the return values from commands. Return values are not displayed on the screen. To get the return value from a command, it should be enclosed in square brackets '[ ]'. Upon execution the return value of the the command will become the value of these brackets. This is called command substitution.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

The commands that produce return values are, but not limited to: **find**, which returns a reference to a particular item, **ping**, which returns the number of sucessful pings, **time**, which returns the measured time value, **incr** and **decr**, which return the new value of a variable, and **add**, which returns the internal number of newly created item.

Example

Consider the usage of **find** command:

```
[admin@MikroTik] > /interface
[admin@MikroTik] interface> find type=ether
[admin@MikroTik] interface>
[admin@MikroTik] interface> :put [find type=ether]
*1,*2
[admin@MikroTik] interface>
```

This way you can see internal console numbers of items. Naturally, you can use them as arguments in other commands:

```
[admin@MikroTik] interface> enable [find type=ether]
[admin@MikroTik] interface>
```

Operators

Description

RouterOS console can do simple calculations with numbers, time values, IP addresses, strings and lists. To get result from an expression with operators, enclose it in parentheses '(' and ')'. The expression result serves as a return value for the parentheses.

Command Description

- - unary minus. Inverts given number value.

- - binary minus. Substracts two numbers, two time values, two IP addresses or an IP address and a number

! - logical NOT. Unary operator, which inverts given boolean value

/ - division. Binary operator. Divides one number by another (gives number) or a time value by a number (gives time value).

. - concatenation. Binary operator, concatenates two string or append one list to another or appends an element to a list.

^ - bitwise XOR. The argumens and the result are both IP addresses

~ - bit inversion. Unary operator, which inverts bits in IP address

***** - multiplication. Binary operator, which can multiply two numbers or a time value by a number.

& - bitwise AND The argumens and the result are both IP addresses

&& - logical AND. Binary operator. The argumens and the result are both logical values

+ - binary plus. Adds two numbers, two time values or a number and an IP address.

< - less. Binary operator which compares two numbers, two time values or two IP addresses. Returns boolean value

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

<< - left shift. Binary operator, which shifts IP address by a given amount of bits. The first argument is an IP address, the second is an integer and the result is an IP address.

<= - less or equal. Binary operator which compares two numbers, two time values or two IP addresses. Returns boolean value

> - greater. Binary operator which compares two numbers, two time values or two IP addresses. Returns boolean value

>= - greater or equal. Binary operator which compares two numbers, two time values or two IP addresses. Returns boolean value

>> - right shift. Binary operator, which shifts IP address by a given amount of bits. The first argument is an IP address, the second is an integer and the result is an IP address.

| - bitwise OR. The argumens and the result are both IP addresses

|| - logical OR. Binary operator. The argumens and the result are both logical values

Notes

When comparing two arrays note, that two arrays are equal only if their respective elements are equal.

Example

Operator priority and evaluation order

```
[admin@MikroTik] ip firewall rule forward> :put (10+1-6*2=11-12=2+(-3)=-1)
false
[admin@MikroTik] ip firewall rule forward> :put (10+1-6*2=11-12=(2+(-3)=-1))
true
[admin@MikroTik] ip firewall rule forward
```

logical NOT

```
[admin@MikroTik] interface> :put (!true)
false
[admin@MikroTik] interface> :put (!(2>3))
true
[admin@MikroTik] interface>
```

unary minus

```
[admin@MikroTik] interface> :put (-1<0)
true
[admin@MikroTik] >
1
```

bit inversion

```
[admin@MikroTik] interface> :put (~255.255.0.0)
0.0.255.255
[admin@MikroTik] interface>
```

sum

```
[admin@MikroTik] interface> :put (3ms + 5s)
00:00:05.003
[admin@MikroTik] interface> :put (10.0.0.15 + 0.0.10.0)
cannot add ip address to ip address
[admin@MikroTik] interface> :put (10.0.0.15 + 10)
10.0.0.25
[admin@MikroTik] interface>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

subtraction

```
[admin@MikroTik] interface> :put (15 - 10)
5
[admin@MikroTik] interface> :put (10.0.0.15 - 10.0.0.3)
12
[admin@MikroTik] interface> :put (10.0.0.15 - 12)
10.0.0.3
[admin@MikroTik] interface> :put (15h - 2s)
14:59:58
[admin@MikroTik] interface>
```

multiplication

```
[admin@MikroTik] interface> :put (12s * 4)
00:00:48
[admin@MikroTik] interface> :put (-5 * -2)
10
[admin@MikroTik] interface>
```

division

```
[admin@MikroTik] interface> :put (10s / 3)
00:00:03.333
[admin@MikroTik] interface> :put (5 / 2)
2
[admin@MikroTik] interface>
[admin@MikroTik] > :put (0:0.10 / 3)
00:00:02
[admin@MikroTik] >
```

comparison

```
[admin@MikroTik] interface> :put (10.0.2.3<=2.0.3.10)
false
[admin@MikroTik] interface> :put (100000s>27h)
true
[admin@MikroTik] interface> :put (60s,1d!=1m,3600s)
true
[admin@MikroTik] interface> :put (bridge=routing)
false
[admin@MikroTik] interface> :put (yes=false)
false
[admin@MikroTik] interface> :put (true=aye)
false
[admin@MikroTik] interface>
```

logical AND, logical OR

```
[admin@MikroTik] interface> :put ((yes && yes) || (yes && no))
true
[admin@MikroTik] interface> :put ((no || no) && (no || yes))
false
[admin@MikroTik] interface>
```

bitwise AND, bitwise OR, bitwise XOR

```
[admin@MikroTik] interface> :put (10.16.0.134 & ~255.255.255.0)
0.0.0.134
[admin@MikroTik] interface>
```

shift operators

```
[admin@MikroTik] interface> :put (~((0.0.0.1 << 7) - 1))
255.255.255.128
[admin@MikroTik] interface>
```

Concatenation

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] interface> :put (1 . 3)
13
[admin@MikroTik] interface> :put (1,2 . 3)
1,2,3
[admin@MikroTik] interface> :put (1 . 3,4)
13,4
[admin@MikroTik] interface> :put (1,2 . 3,4)
1,2,3,4
[admin@MikroTik] interface> :put ((1 . 3) + 1)
14
[admin@MikroTik] interface>
```

Data types

Description

The RouterOS console differentiates between several data types, which are string, boolean, number, time interval, IP address, internal number and list. The console tries to convert any value to the most specific type first, backing if it fails. The order in which the console attempts to convert an entered value is presented below:

- list
- internal number
- number
- IP address
- time
- boolean
- string

Internal scripting language supplies special functions to explicitly control type conversion. The **toarray**, **tobool**, **toid**, **toip**, **tonum**, **tostr** and **totime** functions convert a value accordingly to **list**, **boolean**, **internal number**, **IP address**, **number**, **string** or **time**.

The number type is internally represented as 64 bit signed integer, so the value a number type variable can take is in range from -9223372036854775808 to 9223372036854775807. It is possible to input number value in hexadecimal form, by prefixing it with **0x**, e.g.:

```
[admin@MikroTik] > :global MyVar 0x10
[admin@MikroTik] > :put $MyVar
16
[admin@MikroTik] >
```

Lists are treated as comma separated sequence of values. Putting whitespaces around commas is not recommended, because it might confuse console about words' boundaries.

Boolean values can be either **true** or **false**. Console also accepts **yes** for **true**, and **no** for **false**.

Internal numbers are preceided ***** sign.

Time intervals can be entered either using HH:MM:SS.MS notation, e.g.:

```
[admin@MikroTik] > :put 01:12:1.01
01:12:01.010
[admin@MikroTik] >
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

or as sequence of numbers, optionally followed by letters specifying the units of time measure (**d** dor days, **h** for hours, **m** for minutes, **s** for seconds and **ms** for milliseconds), e.g.:

```
[admin@MikroTik] > :put 2d11h12
2d11:00:12
[admin@MikroTik] >
```

As can bee seen, time values with omitted unit specificators are treated as expressed in seconds.

- **d, day, days** - one day, or 24 hours
- **h, hour, hours** - one hour
- **m, min** - one minute
- **s** - one second
- **ms** - one millisecond, id est 0.001 second

Possible aliases for time units:

The console also accepts time values with decimal point:

```
[admin@MikroTik] > :put 0.1day1.2s
02:24:01.200
[admin@MikroTik] >
```

Command Reference

Description

RouterOS has a number of built-in console commands and expressions (ICE) that do not depend on the current menu level. These commands do not change configuration directly, but they are useful for automating various maintenance tasks. The full ICE list can be accessed by typing '?' after the ':' prefix (therefore it can be safely assumed that all ICE have ':' prefix), for example:

```
[admin@MikroTik] > :
beep   environment  for      if   list   pick     set      tobool  tonum   typeof
delay  execute      foreach  led  local  put      time     toid    tostr   unset
do     find         global   len  log    resolve  toarray  toip    totime  while
[admin@MikroTik] >
```

Command Description

beep - forces the built-in PC beeper to produce a signal for length seconds at frequency Hz. (*integer*; default: **1000**) - signal frequency measured in Hz (*time*; default: **100ms**) - signal length

```
[admin@MikroTik] > :beep length=2s frequency=10000
[admin@MikroTik] >
```

delay - does nothing for a given amount of time. (*time*) - amount of time to wait

- **omitted** - delay forever

do - executes commands repeatedly until given conditions are met. If no parameters are given, do just executes its payload once, which does not make much use. If a logical condition is specified for the while parameter, it will be evaluated after executing commands, and in case it is true, do statement is executed again and again until false. The if parameter, if present, is evaluated only once before doing anything else, and if it is false then no action is taken (*text*) - actions to execute

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

repeatedly (yes | no) - condition, which is evaluated each time after the execution of enclosed statements (yes | no) - condition, which is evaluated once before the execution of enclosed statements

```
[admin@MikroTik] > {:global i 10; :do {:put $i; :set i ($i - 1);} \
\... while (($i < 11) && ($i > 0)); :unset i;}
10
9
8
7
6
5
4
3
2
1
[admin@MikroTik] >
```

environment print - prints information about variables that are currently initialised. All global variables in the system are listed under the heading Global Variables. All variables that are introduced in the current script (variables introduced by :local or created by :for or :foreach statements) are listed under the heading Local Variables.

Creating variables and displaying a list of them

```
[admin@MikroTik] > :local A "This is a local variable"
[admin@MikroTik] > :global B "This is a global one"
[admin@MikroTik] > :environment print
Global Variables
B=This is a global one
Local Variables
A=This is a local variable
[admin@MikroTik] >
```

find - searches for substring inside a string or for an element with particular value inside an array, depending on argument types and returns position at which the value is found. The elements in list and characters in string are numbered from 0 upwards (*text* |) - the string or value list the search will be performed in (*text*) - value to be searched for (*integer*) - position after which the search is started

```
[admin@MikroTik] interface pppoe-server> :put [:find "13sdf1sdfss1sfsdf324333" ]
0
[admin@MikroTik] interface pppoe-server> :put [:find "13sdf1sdfss1sfsdf324333" 3 ]
1
[admin@MikroTik] interface pppoe-server> :put [:find "13sdf1sdfss1sfsdf324333" 3 3]
17
[admin@MikroTik] interface pppoe-server> :put [:find "1,1,1,2,3,3,4,5,6,7,8,9,0,1,2,3"
3 ]
4
[admin@MikroTik] interface pppoe-server> :put [:find "1,1,1,2,3,3,4,5,6,7,8,9,0,1,2,3"
3 3]
4
[admin@MikroTik] interface pppoe-server> :put [:find "1,1,1,2,3,3,4,5,6,7,8,9,0,1,2,3"
3 4]
5
[admin@MikroTik] interface pppoe-server> :put [:find "1,1,1,2,3,3,4,5,6,7,8,9,0,1,2,3"
3 5]
15
[admin@MikroTik]
```

for - executes supplied commands over a given number of iterations, which is explicity set through from and to parameters (*name*) - the name of the loop counter variable (*integer*) - start value of the loop counter variable (*integer*) - end value of the loop counter variable (*integer*; default: **1**) - increment value. Depending on the loop counter variable start and end values, step parameter can be treated also as decrement (*text*) - contains the command to be executed repeatedly

```
[admin@MikroTik] > :for i from=1 to=100 step=37 do={:put ($i . " - " . 1000/$i)}
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
1                                       -                                       1000
38                                       -                                        26
75                                       -                                        13
[admin@MikroTik] >
```

foreach - executes supplied commands for each element in list (*name*) - the name of the loop counter variable () - list of values over which to iterate (*text*) - contains the command to be executed repeatedly

Printing a list of available interfaces with their respective IP addresses

```
:foreach        i        in=[/interface        find        type=ether        ]        \
\...     do={:put     ("+--"     .     [/interface     get     $i     name]);     \
\...      :foreach      j      in=[/ip      address      find      interface=$i]
\...    do={:put    ("|     `--"    .    [/ip    address    get    $j    address])}}
+--ether1
|                                                                        `--1.1.1.3/24
|                                                                    `--192.168.50.1/24
|                                                                        `--10.0.0.2/24
+--ether2
|                                                                       `--10.10.0.2/24
[admin@MikroTik] >
```

global - declares global variable (*name*) - name of the variable (*text*) - value, which should be assigned to the variable

```
[admin@MikroTik]      >      :global      MyString      "This      is      a      string"
[admin@MikroTik]            >            :global            IPAddr            10.0.0.1
[admin@MikroTik]             >             :global             time              0:10
[admin@MikroTik]                  >                 :environment                print
Global                                                                    Variables
IPAddr=10.0.0.1
time=00:10:00
MyString=This                      is                      a                     string
Local                                                                     Variables
[admin@MikroTik] >
```

if - conditional statement. If a given logical condition evaluates to true then the do block of commands is executed. Otherwice an optional else block is executed. (yes | no) - logical condition, which is evaluated once before the execution of enclosed statements (*text*) - this block of commands is executed if the logical condition evaluates to true (*text*) - this block of commands is executed if the logical condition evaluates to false

Check if the firewall has any rules added

```
[admin@MikroTik]  >  :if  ([:len  [/ip  firewall  filter  find]]  >  0)  do={:put  true}
else={:put                                                                      false}
true
[admin@MikroTik] >
```

Check whether the gateway is reachable. In this example, the IP address of the gateway is **10.0.0.254**

```
[admin@MikroTik] > :if ([/ping 10.0.0.254 count=1] = 0) do {:put "gateway unreachable"}
10.0.0.254                              ping                                    timeout
1    packets    transmitted,    0    packets    received,    100%    packet    loss
gateway                                                                  unreachable
[admin@MikroTik] >
```

led - allows to control system built-in LEDs (Light Emitting Diodes). This command is available only on RouterBoard platforms whith the routerboard or rb500 packages installed, depending on the RouterBoard model. The number of available LEDs varies between different RouterBoard models (yes | no) - controls first LED (yes | no) - controls second LED (yes | no) - controls third LED (yes | no) - controls fourth LED (*time*) - specifies the length of the action

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **omitted** - altar LED state forever

Switch on LEDs 2 and 3 for 5 seconds

```
[admin@MikroTik] > :led led2=yes led3=yes length=5s
```

len - returns the number of characters in string or the number of elements in list depending on the type of the argument (*name*) - string or list the length of which should be returned

```
[admin@MikroTik]          >          :put          [:len          gvejimezyfopmekun]
17
[admin@MikroTik]         >         :put         [:len         gve,jim,ezy,fop,mek,un]
6
[admin@MikroTik] >
```

list - displays a list of all available console commands that match given search key(s) (*text*) - first search key (*text*) - second search key (*text*) - third search key

Display console commands that have **hotspot**, **add** and **user** parts in the command's name and path

```
[admin@MikroTik]        >        :list        user        hotspot        "add        "
List  of  console  commands  under  "/"  matching  "user"  and  "hotspot"  and  "add  ":

ip     hotspot     profile     add     name=     hotspot-address=     dns-name=     \
\...       html-directory=       rate-limit=       http-proxy=       smtp-server=       \
\...    login-by=    http-cookie-lifetime=    ssl-certificate=    split-user-domain=    \
\...      use-radius=      radius-accounting=      radius-interim-update=      copy-from=
ip   hotspot   user   add   server=   name=   password=   address=   mac-address=   \
\...    profile=    routes=    limit-uptime=    limit-bytes-in=    limit-bytes-out=    \
\...                copy-from=                  comment=                  disabled=
ip   hotspot    user    profile    add    name=    address-pool=    session-timeout=    \
\...        idle-timeout=        keepalive-timeout=        status-autorefresh=        \
\...     shared-users=     rate-limit=     incoming-filter=     outgoing-filter=     \
\...  incoming-mark=  outgoing-mark=  open-status-page=  on-login=  on-logout=  copy-from=
[admin@MikroTik] >
```

local - declares local variable (*name*) - name of the variable (*text*) - value, which should be assigned to the variable

```
[admin@MikroTik]      >      :local      MyString      "This      is      a      string"
[admin@MikroTik]             >             :local             IPAddr             10.0.0.1
[admin@MikroTik]              >              :local              time              0:10
[admin@MikroTik]                  >                  :environment                  print
Global                                                                  Variables
Local                                                                   Variables
IPAddr=10.0.0.1
time=00:10:00
MyString=This                     is                     a                     string
[admin@MikroTik] >
```

log - adds a message specified by message parameter to the system logs. (*name*) - name of the logging facility to send message to (*text*) - the text of the message to be logged

Send message to **info** log

```
[admin@MikroTik] > :log info "Very Good thing happened. We have received our first
packet!"
[admin@MikroTik]             >             /log             print             follow
...
19:57:46 script,info Very Good thing happened. We have received our first packet!
...
```

pick - returns a range of elements or a substring depending on the type of input value (*text* |) - the string or value list from which a substring or a subrange should be returned (*integer*) - start position of substring or subrange (*integer*) - end position for substring or subrange

```
[admin@MikroTik]            >            :set            a            1,2,3,4,5,6,7,8
[admin@MikroTik]              >              :put              [:len              $a]
8
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik]          >          :put          [:pick          $a]
1
[admin@MikroTik]         >         :put         [:pick         $a         0         4]
1,2,3,4
[admin@MikroTik]        >        :put        [:pick        $a        2        4]
3,4
[admin@MikroTik]         >         :put         [:pick         $a         2]
3
[admin@MikroTik]       >       :put       [:pick       $a       5       1000000]
6,7,8
[admin@MikroTik]         >         :set         a         abcdefghij
[admin@MikroTik]         >         :put         [:len         $a]
10
[admin@MikroTik]         >         :put         [:pick         $a]
a
[admin@MikroTik]       >       :put       [:pick       $a       0       4]
abcd
[admin@MikroTik]       >       :put       [:pick       $a       2       4]
cd
[admin@MikroTik]        >        :put        [:pick        $a        2]
c
[admin@MikroTik]        >        :put        [:pick        $a        5        1000000]
fghij
```

put - echoes supplied argument to the console (*text*) - the text to be echoed to the console

Display the MTU of **ether1** interface

```
[admin@MikroTik]       >       :put       [/interface       get       ether1       mtu]
1500
[admin@MikroTik] >
```

resolve - returns IP address of the host resolved from the DNS name. The DNS settings should be configured on the router (/ip dns submenu) prior to using this command. (*text*) - domain name to be resolved into an IP address

DNS configuration and **resolve** command example

```
[admin@MikroTik]     ip     route>     /ip     dns     set     primary-dns=159.148.60.2
[admin@MikroTik]       ip       route>       :put       [:resolve       www.example.com]
192.0.34.166
```

set - assigns new value to a variable (*name*) - the name of the variable (*text*) - the new value of the variable

Measuring time needed to resolve www.example.com

```
[admin@MikroTik]      >      :put      [:time      [:resolve      www.example.com      ]]
00:00:00.006
[admin@MikroTik] >
```

time - measures the amount of time needed to execute given console commands (*text*) - the console commands to measure execution time of

Measuring time needed to resolve www.example.com

```
[admin@MikroTik]      >      :put      [:time      [:resolve      www.example.com      ]]
00:00:00.006
[admin@MikroTik] >
```

while - executes given console commands repeatedly while the logical conditions is true (yes | no) - condition, which is evaluated each time before the execution of enclosed statements (*text*) - console commands that should be executed repeatedly

```
[admin@MikroTik] > :set i 0; :while ($i < 10) do={:put $i; :set i ($i + 1)};
0
1
2
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
3
4
5
6
7
8
9
[admin@MikroTik] >
```

Special Commands

Description

Monitor

It is possible to access values that are shown by most **monitor** actions from scripts. A **monitor** command that has a **do** parameter can be supplied either script name (see **/system scripts**), or console commands to execute.

Get

Most **print** commands produce values that are accessible from scripts. Such **print** commands have a corresponding **get** command on the same menu level. The **get** command accepts one parameter when working with regular values or two parameters when working with lists.

Notes

Monitor command with **do** argument can also be called directly from scripts. It will not print anything then, just execute the given script.

The names of the properties that can be accessed by **get** are the same as shown by **print** command, plus names of item flags (like the disabled in the example below). You can use [T ab] key completions to see what properties any particular **get** action can return.

Example

In the example below **monitor** action will execute given script each time it prints stats on the screen, and it will assign all printed values to local variables with the same name:

```
[admin@MikroTik] interface> monitor-traffic ether2 once do={:environment print}
    received-packets-per-second: 0
       received-bits-per-second: 0bps
        sent-packets-per-second: 0
           sent-bits-per-second: 0bps

Global Variables
i=1
Local Variables
sent-bits-per-second=0
received-packets-per-second=0
received-bits-per-second=0
sent-packets-per-second=0
[admin@MikroTik] interface>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Additional Features

Description

To include comment in the console script prefix it with '#'. In a line of script that starts with '#' all characters until the newline character are ignored.

To put multiple commands on a single line separate them with ';'. Console treats ';' as the end of line in scripts.

Any of the **{}[]"'\$** characters should be escaped in a regular string with '\' character. Console takes any character following '\' literally, without assigning any special meaning to it, except for such cases:

```
\a      bell (alarm), character code 7
\b      backspace, character code 8
\f      form feed, character code 12
\n      newline, character code 10
\r      carriage return, character code 13
\t      tabulation, character code 9
\v      vertical tabulation, character code 11
\_      space, character code 32
```

Note that '\', followed by any amount of whitespace characters (spaces, newlines, carriage returns, tabulations), followed by newline is treated as a single whitespace, except inside quotes, where it is treated as nothing. This is used by console to break up long lines in scripts generated by export commands.

Script Repository

Home menu level: ***/system script***

Description

All scripts are stored in the **/system script** menu along with some service information such as script name, script owner, number of times the script was executed and permissions for particular script.

In RouterOS, a script may be automatically started in three different ways:

- via the scheduler
- on event occurence - for example, the netwatch tool generates an event if a network host it is configured to monitor becomes unaccessible
- by another script

It is also possible to start a script manually via **/system script run** command.

Property Description

last-started (*time*) - date and time when the script has been last invoked. The argument is shown only if the run-count!=0.

owner (*name*; default: **admin**) - the name of the user who created the script

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

policy (*multiple choice: ftp | local | policy | read | reboot | ssh | telnet | test | web | write*; default: **reboot,read,write,policy,test**) - the list of the policies applicable:

- **ftp** - user can log on remotely via ftp and send and retrieve files from the router
- **local** - user can log on locally via console
- **policy** - manage user policies, add and remove user
- **read** - user can retrieve the configuration
- **reboot** - user can reboot the router
- **ssh** - user can log on remotely via secure shell
- **telnet** - user can log on remotely via telnet
- **test** - user can run ping, traceroute, bandwidth test
- **web** - user can log on remotely via http
- **write** - user can retrieve and change the configuration

run-count (*integer*; default: **0**) - script usage counter. This counter is incremented each time the script is executed. The counter will reset after reboot.

source (*text*; default: **""**) - the script source code itself

Command Description

run (*name*) - executes a given script (*name*) - the name of the script to execute

Notes

You cannot do more in scripts than you are allowed to do by your current user rights, that is, you cannot use disabled policies. For example, if there is a policy group in **/user group** which allows you **ssh,local,telnet,read,write,policy,test,web** and this group is assigned to your user name, then you cannot make a script that reboots the router.

Example

The following example is a script for writing message "Hello World!" to the **info** log:

```
[admin@MikroTik] system script> add name="log-test" source={:log info "Hello World!"}
[admin@MikroTik] system script> run log-test

[admin@MikroTik] system script> print
 0 name="log-test" owner="admin"
policy=ftp,reboot,read,write,policy,test,winbox,password last-started=mar/20/2001
22:51:41
   run-count=1 source=:log info "Hello World!"
[admin@MikroTik] system script>
```

Task Management

Home menu level: ***/system script job***

Description

This facility is used to manage the active or scheduled tasks.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Property Description

name (*read-only: name*) - the name of the script to be referenced when invoking it

owner (*text*) - the name of the user who created the script

source (*read-only: text*) - the script source code itself

Example

```
[admin@MikroTik] system script> job print
  # SCRIPT  OWNER                   STARTED
  0 DelayeD admin                   dec/27/2003 11:17:33

[admin@MikroTik] system script>
```

You can cancel execution of a script by removing it from the job list

```
[admin@MikroTik] system script> job remove 0
[admin@MikroTik] system script> job print

[admin@MikroTik] system script>
```

Script Editor

Command name: ***/system script edit***

Description

RouterOS console has a simple full-screen editor for scripts with support for multiline script writing.

Keyboard Shortcuts

- **Delete** - deletes character at cursor position
- **Ctrl+h, backspase** - deletes character before cursor. Unindents line
- **Tab** - indents line
- **Ctrl+b, LeftArrow** - moves cursor left
- **Ctrl+f, RightArrow** - moves cursor right
- **Ctrl+p, UpArrow** - moves cursor up
- **Ctrl+n, DownArrow** - moves cursor down
- **Ctrl+a, Home** - moves cursor to the beginning of line or script
- **Ctrl+e, End** - moves cursor to the end of line or script
- **Ctrl+y** - inserts contents of buffer at cursor position
- **Ctrl+k** - deletes characters from cursor position to the end of line
- **Ctrl+u** - undoes last action
- **Ctrl+o** - exits editor accepting changes
- **Ctrl+x** - exits editor discarding changes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Command Description

edit (*name*) - opens the script specified by the name argument in full-screen editor

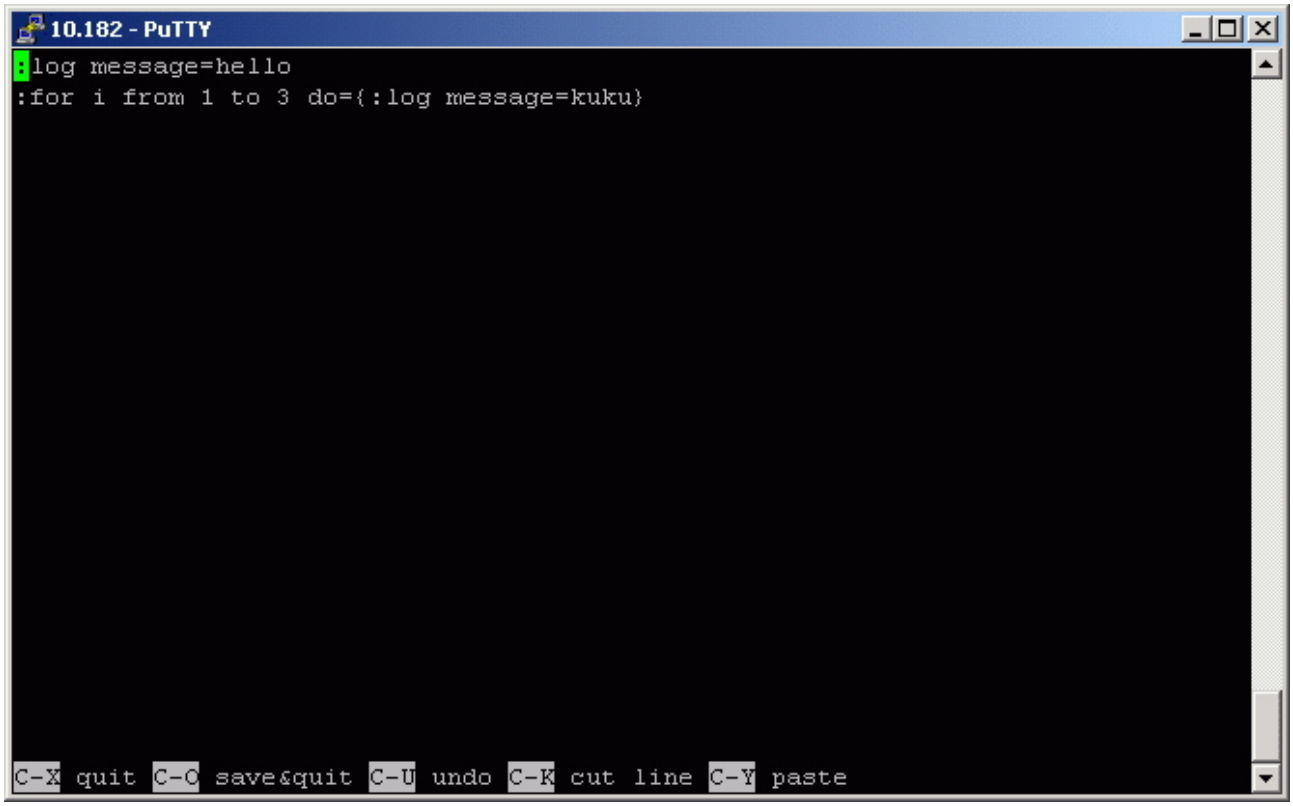
Notes

All characters that are deleted by **backspace, delete** or **Ctrl+k** keys are accumulated in the buffer. Pressing any other key finishes adding to this buffer (**Ctrl+y** can paste it's contents), and the next delete operation will replace it's contents. Undo doesn't change contents of cut buffer.

Script editor works only on VT102 compatible terminals (terminal names "vt102", "linux", "xterm", "rxvt" are recognized as VT102 at the moment). Delete, backspace and cursor keys might not work with all terminal programs, use 'Ctrl' alternatives in such cases.

Example

The following example shows the script editor window with a sample script open:



This script is used for writing message "hello" and 3 messages "kuku" to the system log.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Scheduler

Document revision 0.9 (Wed Nov 24 12:48:55 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
Scheduler Configuration
  Description
  Property Description
  Notes
  Example

General Information

Summary

System Scheduler executes scripts at designated time.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/system scheduler***
Standards and Technologies: ***None***
Hardware usage: ***Not significant***

Related Documents

- ***Package Management***
- ***Scripting Examples***
- ***Scripting Examples***

Scheduler Configuration

Description

The scheduler can trigger script execution at a particular time moment, after a specified time interval, or both.

Property Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

interval (*time*; default: **0s**) - interval between two script executions, if time interval is set to zero, the script is only executed at its start time, otherwise it is executed repeatedly at the time interval is specified

name (*name*) - name of the task

on-event (*name*) - name of the script to execute. It must be presented at /system script

run-count (*read-only: integer*) - to monitor script usage, this counter is incremented each time the script is executed

start-date (*date*) - date of the first script execution

start-time (*time*) - time of the first script execution

Notes

Rebooting the router will reset **run-count** counter.

If more than one script has to be executed simultaneously, they are executed in the order they appear in the scheduler configuration. This can be important if one scheduled script is used to disable another one. The order of scripts can be changed with the **move** command.

If a more complex execution pattern is needed, it can usually be done by scheduling several scripts, and making them enable and disable each other.

Example

We will add a task that executes the script **log-test** every hour:

```
[admin@MikroTik] system script> add name=log-test source=:log message=test
[admin@MikroTik] system script> print
    0 name="log-test" source=":log messgae=test" owner=admin run-count=0
[admin@MikroTik] system script> .. scheduler
[admin@MikroTik] system scheduler> add name=run-1h interval=1h
on-event=log-test
[admin@MikroTik] system scheduler> print

Flags: X - disabled
 #   NAME      ON-EVENT  START-DATE  START-TIME INTERVAL     RUN-COUNT
 0   run-1h    log-test  mar/30/2004 06:11:35   1h           0
[admin@MikroTik] system scheduler>
```

In another example there will be two scripts added that will change the bandwidth setting of a queue rule "Cust0". Every day at 9AM the queue will be set to 64Kb/s and at 5PM the queue will be set to 128Kb/s. The queue rule, the scripts, and the scheduler tasks are below:

```
[admin@MikroTik] queue simple> add name=Cust0 interface=ether1 \
\... dst-address=192.168.0.0/24 limit-at=64000
[admin@MikroTik] queue simple> print
Flags: X - disabled, I - invalid
  0   name="Cust0" target-address=0.0.0.0/0 dst-address=192.168.0.0/24
      interface=ether1 limit-at=64000 queue=default priority=8 bounded=yes

[admin@MikroTik] queue simple> /system script
[admin@MikroTik] system script> add name=start_limit source={/queue simple set \
\... Cust0 limit-at=64000}
[admin@MikroTik] system script> add name=stop_limit source={/queue simple set \
\... Cust0 limit-at=128000}
[admin@MikroTik] system script> print
  0 name="start_limit" source="/queue simple set Cust0 limit-at=64000"
    owner=admin run-count=0

  1 name="stop_limit" source="/queue simple set Cust0 limit-at=128000"
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
    owner=admin run-count=0

[admin@MikroTik] system script> .. scheduler
[admin@MikroTik] system scheduler> add interval=24h name="set-64k" \
\... start-time=9:00:00 on-event=start_limit
[admin@MikroTik] system scheduler> add interval=24h name="set-128k" \
\... start-time=17:00:00 on-event=stop_limit
[admin@MikroTik] system scheduler> print
Flags: X - disabled
  #   NAME      ON-EVENT START-DATE  START-TIME INTERVAL            RUN-COUNT
  0   set-64k   start... oct/30/2008 09:00:00   1d                  0
  1   set-128k  stop_... oct/30/2008 17:00:00   1d                  0
[admin@MikroTik] system scheduler>
```

The following example schedules a script that sends each week a backup of router configuration by e-mail.

```
[admin@MikroTik] system script> add name=e-backup source={/system backup
{... save name=email; /tool e-mail send to="root@host.com" subject=([/system
{... identity get name] . " Backup") file=email.backup}
[admin@MikroTik] system script> print
  0 name="e-backup" source="/system backup save name=ema... owner=admin
    run-count=0

[admin@MikroTik] system script> .. scheduler
[admin@MikroTik] system scheduler> add interval=7d name="email-backup" \
\... on-event=e-backup
[admin@MikroTik] system scheduler> print
Flags: X - disabled
  #   NAME      ON-EVENT START-DATE  START-TIME INTERVAL            RUN-COUNT
  0   email-... e-backup oct/30/2008 15:19:28   7d                  1
[admin@MikroTik] system scheduler>
```

Do not forget to set the e-mail settings, i.e., the SMTP server and From: address under **/tool e-mail**. For example:

```
[admin@MikroTik] tool e-mail> set server=159.148.147.198 from=SysAdmin@host.com
[admin@MikroTik] tool e-mail> print
    server: 159.148.147.198
      from: SysAdmin@host.com
[admin@MikroTik] tool e-mail>
```

Example below will put 'x' in logs each hour from midnight till noon:

```
[admin@MikroTik] system script> add name=enable-x source={/system scheduler
{... enable x}
[admin@MikroTik] system script> add name=disable-x source={/system scheduler
{... disable x}
[admin@MikroTik] system script> add name=log-x source={:log message=x}
[admin@MikroTik] system script> .. scheduler
[admin@MikroTik] system scheduler> add name=x-up start-time=00:00:00 \
\... interval=24h on-event=enable-x
[admin@MikroTik] system scheduler> add name=x-down start-time=12:00:00
\... interval=24h on-event=disable-x
[admin@MikroTik] system scheduler> add name=x start-time=00:00:00 interval=1h \
\... on-event=log-x
[admin@MikroTik] system scheduler> print
Flags: X - disabled
  #   NAME      ON-EVENT START-DATE  START-TIME INTERVAL            RUN-COUNT
  0   x-up      enable-x oct/30/2008 00:00:00   1d                  0
  1   x-down    disab... oct/30/2008 12:00:00   1d                  0
  2   x         log-x    oct/30/2008 00:00:00   1h                  0
[admin@MikroTik] system scheduler>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Traffic Monitor

Document revision 1 (Thu Jul 07 08:34:34 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
Traffic Monitor
  Description
  Property Description
  Example

General Information

Summary

Traffic monitor executes scripts on a specific datarate throough an interface.

Specifications

Packages required: ***advanced-tools***
License required: ***level1***
Home menu level: ***/tool traffic-monitor***
Standards and Technologies: ***none***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***
- ***Scripting Host***

Traffic Monitor

Home menu level: ***/tool traffic-monitor***

Description

The traffic monitor tool is used to execute console scripts when interface traffic crosses a given threshold. Each item in traffic monitor list consists of its name (which is useful if you want to disable or change properties of this item from another script), some parameters, specifying traffic condition, and the pointer to a script or scheduled event to execute when this condition is met.

Property Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

interface (*name*) - interface to monitor

name (*name*) - name of the traffic monitor item

on-event (*name*) - script source. Must be present under /system script

threshold (*integer*; default: **0**) - traffic threshold

traffic (*transmitted* | *received*; default: **transmitted**) - type of traffic to monitor

- **transmitted** - transmitted traffic
- **received** - received traffic

trigger (*above* | *always* | *below*; default: **above**) - condition on which to execute the script

- **above** - the script will be run each time the traffic exceeds the threshold
- **always** - triggers scripts on both - above and below condition
- **below** - triggers script in the opposite condition, when traffic reaches a value that is lower than the threshold

Example

In this example the traffic monitor enables the interface ether2, if the received treffic exceeds 15kbps on ether1, and disables the interface ether2, if the received traffic falls below 12kbps on ether1.

```
[admin@MikroTik] system script> add name=eth-up source={/interface enable ether2}
[admin@MikroTik] system script> add name=eth-down source={/interface disable
{... ether2}
[admin@MikroTik] system script> /tool traffic-monitor
[admin@MikroTik] tool traffic-monitor> add name=turn_on interface=ether1 \
\... on-event=eth-up threshold=15000 trigger=above traffic=received
[admin@MikroTik] tool traffic-monitor> add name=turn_off interface=ether1 \
\... on-event=eth-down threshold=12000 trigger=below traffic=received
[admin@MikroTik] tool traffic-monitor> print
Flags: X - disabled, I - invalid
  #   NAME           INTERFACE     TRAFFIC     TRIGGER THRESHOLD  ON-EVENT
  0   turn_on        ether1        received    above   15000      eth-up
  1   turn_off       ether1        received    below   12000      eth-down
[admin@MikroTik] tool traffic-monitor>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

IP Telephony

Document revision 2.2 (Mon Apr 26 12:53:19 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
- Summary
- Specifications
- Related Documents
- Description
- Notes
- Additional Documents

General Voice port settings
- Description
- Property Description
- Notes

Voicetronix Voice Ports
- Property Description
- Command Description
- Notes

LineJack Voice Ports
- Property Description
- Command Description
- Notes

PhoneJack Voice Ports
- Property Description
- Command Description

Zaptel Voice Ports
- Property Description
- Command Description

ISDN Voice Ports
- Property Description
- Command Description
- Notes

Voice Port for Voice over IP (voip)
- Description
- Property Description

Numbers
- Description
- Property Description
- Notes
- Example

Regional Settings
- Description
- Property Description
- Notes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Audio CODECs
  Description
  Example
AAA
  Description
  Property Description
  Notes
Gatekeeper
  Description
  Property Description
  Example
  Example
Troubleshooting
  Description
A simple example
  Description
  Setting up the MikroTik IP Telephone
  Setting up the IP Telephony Gateway
  Setting up the Welltech IP Telephone
  Setting up MikroTik Router and CISCO Router
  Setting up PBX to PBX Connection over an IP Network

General Information

Summary

The MikroTik RouterOS IP Telephony feature enables Voice over IP (VoIP) communications using routers equipped with the following voice port hardware:

- Quicknet LineJACK or PhoneJACK analog telephony cards
- ISDN cards
- Voicetronix OpenLine4 (was V4PCI) - 4 analog telephone lines cards
- Zaptel Wildcard X100P IP telephony card - 1 analog telephone line

Specifications

Packages required: ***telephony***
License required: ***level1***
Home menu level: ***/ip telephony***
Standards and Technologies: ***RTP***
Hardware usage: *Pentium MMX level processor recommended*

Related Documents

- ***Package Management***
- ***ISDN***
- ***AAA***

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Description

IP telephony, known as Voice over IP (VoIP), is the transmission of telephone calls over a data network like one of the many networks that make up the Internet. There are four ways that you might talk to someone using VoIP:

- Computer-to-computer - This is certainly the easiest way to use VoIP, and you don't have to pay for long-distance calls.
- Computer-to-telephone - This method allows you to call anyone (who has a phone) from your computer. Like computer-to-computer calling, it requires a software client. The software is typically free, but the calls may have a small per-minute charge.
- Telephone-to-computer - Allows a standard telephone user to initiate a call to a computer user.
- Telephone-to-telephone - Through the use of gateways, you can connect directly with any other standard telephone in the world.

Suppoted hardware:

- ***Quicknet Technologies*** cards:
  - Internet PhoneJACK (ISA or PCI) for connecting an analog telephone (FXS port)
  - Internet LineJACK (ISA) for connecting an analog telephone line (FXO port) or a telephone (FXS port)
- ISDN client cards (PCI) for connecting an ISDN line. See ***Device Driver List*** for the list of supported PCI ISDN cards
- ***Voicetronix*** OpenLine4 card for connecting four (4) analog telephone lines (FXO ports)
- Zaptel Wildcard X100P IP telephony card (from ***Linux Support Services***) for connecting one analog telephone line (FXO port)

Supported standards:

- MikroTik RouterOS supports IP Telephony in compliance with the International Telecommunications Union - Telecommunications (ITU-T) specification H.323v4. H.323 is a specification for transmitting multimedia (voice, video, and data) across an IP network. H.323v4 includes: H.245, H.225, Q.931, H.450.1, RTP(real-time protocol)
- The followong audio codecs are supported: **G.711** (the 64 kbps Pulse code modulation (PCM) voice coding), **G.723.1** (the 6.3 kbps compression technique that can be used for compressing audio signal at very low bit rate), **GSM-06.10** (the 13.2 kbps coding), **LPC-10** (the 2.5 kbps coding), **G.729** and **G.729a** (the 8 kbps CS-ACELP software coding), **G.728** (16 kbps coding technique, supported only on Quicknet LineJACK cards)

In PSTN lines there is a known delay of the signal caused by switching and signal compressing devices of the telephone network (so, it depends on the distance between the peers), which is generally rather low. The delay is also present in IP networks. The main difference between a PSTN and an IP network is that in IP networks that delay is more random. The actual packet delay may vary in order of magnutude in congested networks (if a network becomes congested, some packets may even be lost). Also packet reordering may take place. To prevent signal loss, caused by random jitter of IP networks and packet reordering, to corrupt audio signal, a jitter buffer is present in IP telephony devices. The jitter buffer is delaying the actual playback of a received packet forming

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

The larger the jitter buffer, the larger the total delay, but fewer packets get lost due to timeout.

The total delay from the moment of recording the voice signal till its playback is the sum of following three delay times:

- delay time at the recording point (approx. 38ms)
- delay time of the IP network (1..5ms and up)
- delay time at the playback point (the jitter delay)

Notes

Each installed Quicknet card requires IO memory range in the following sequence: the first card occupies addresses 0x300-0x31f, the second card 0x320-0x33f, the third 0x340-0x35f, and so on. Make sure there is no conflict in these ranges with other devices, e.g., network interface cards, etc.

Use the telephony logging feature to debug your setup.

Additional Documents

General Voice port settings

Home menu level: ***/ip telephony voice-port***

Description

This submenu is used for managing all IP telephony voice ports (linejack, phonejack, isdn, voip, voicetronix, zaptel)

Property Description

name (*name*) - assigned name of the voice port

type (*read-only: phonejack* | *linejack* | *phonejack-lite* | *phonejack-pci* | *voip* | *isdn* | *voicetronix* | *zaptel*) - type of the installed telephony voice port:

- **phonejack** - Quicknet PhoneJACK (ISA)
- **linejack** - Quicknet LineJACK (ISA)
- **phonejack-lite** - Quicknet PhoneJACK Lite Linux Edition (ISA)
- **phonejack-pci** - Quicknet PhoneJACK (PCI)
- **voip** - generic Voice over IP port
- **isdn** - ISDN cards
- **voicetronix** - Voicetronix OpenLine4
- **zaptel** - Zaptel Wildcard X100P

autodial (*integer*; default: "") - number to be dialed automatically, if call is coming in from this voice port

Notes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

If **autodial** does not exactly match an item in **/ip telephony numbers**, there can be two possibilities:

- if **autodial** is incomplete, rest of the number is asked (local voice port) or incoming call is denied (VoIP)
- if **autodial** is invalid, line is hung up (PSTN line), busy tone is played (POTS) or incoming call is denied (VoIP)

Voicetronix Voice Ports

Home menu level: ***/ip telephony voice-port voicetronix***

Property Description

name (*name*) - name given by the user or the default one

autodial (*integer*; default: **""**) - phone number which will be dialed immediately after the handset has been lifted. If this number is incomplete, then the remaining part has to be dialed on the dial-pad. If the number is incorrect, the line is hung up. If the number is correct, then the appropriate number is dialed (the direct-call mode is used - the line is picked up only after the remote party answers the call)

playback-volume (*integer*: -48..48; default: **0**) - playback volume in dB

- **0** - 0dB meand no change to signal level

record-volume (*integer*: -48..48; default: **0**) - record volume in dB

- **0** - 0dB meand no change to signal level

region (*name*; default: **us**) - regional setting for the voice port. This setting is used for setting the parameters of PSTN line, as well as for detecting and generating the tones

agc-on-playback (yes | no; default: **no**) - automatic gain control on playback (can not be used together with hardware voice codecs)

agc-on-record (yes | no; default: **no**) - automatic gain control on record (can not be used together with hardware voice codecs)

detect-cpt (yes | no; default: **no**) - automatically detect call progress tones

balance-registers (*integer*: 0..255; default: **199**) - registers which depend on telephone line impedance. Can be adjusted to get best echo cancellation. Should be changed only if echo cancellation on voicetronix card does not work good enough. Echo cancellation problems can imply DTMF and busy-tone detection failures. The value has to be in format bal1[,bal3[,bal2]], where bal1, bal2, bal3 - balance registers. bal1 has to be in interval 192..248 (0xC0..0xF8). The others should be in interval 0..255 (0x00..0xFF)

balance-status (*read-only: integer*; default: **unknown**) - shows quality of hardware echo cancellation in dB

loop-drop-detection (yes | no; default: **yes**) - automatically clear call when loop drop is detected

Command Description

test-balance - current balance-registers value is tested once. Result is placed in balance-status parameter. Balance can be tested only when line is off-hook. It won't work if line is on-hook or there is an established connection (*name*) - port name to test balance of

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

find-best-balance - series of test-balance is executed with different balance-registers values. During the tests balance-registers are updated to the best values found (*name*) - port name to find best balance of

clear-call - terminate a current call established with the specified voice port (*name*) - port name to clear call with

show-stats - show voice port statistics (*name*) - port name show statistics of (*time*) - maximal time of packet round trip (*integer*) - number of packets sent by this card (these packets are digitalized input of the voice port) (*integer*) - number of bytes sent by this card (these packets are digitalized input of the voice port) (*text*) - minimal/average/maximal intervals between packets sent (*integer*) - number of packets received by this card (these packets form analog output of the voice port) (*integer*) - number of bytes received by this card (these packets form analog output of the voice port) (*text*) - minimal/average/maximal intervals between packets received (*time*) - approximate delay time from the moment of receiving an audio packet from the IP network till it is played back over the telephony voice port. The value shown is never less than 30ms, although the actual delay time could be less. If the shown value is >40ms, then it is close (+/-1ms) to the actual delay time.

monitor - monitor status of the voice port (*name*) - port name to monitor (*on-hook* | *off-hook* | *ring* | *connection* | *busy*) - current state of the port:

- **on-hook** - the handset is on-hook, no activity
- **off-hook** - the handset is off-hook, the number is being dialed
- **ring** - call in progress, direction of the call is shown by the direction property
- **connection** - the connection has been established
- **busy** - the connection has been terminated, the handset is still off-hook

(*ip-to-port* | *port-to-ip*) - direction of the call

- **ip-to-port** - call from the IP network to the voice card
- **port-to-ip** - call from the voice card to an IP address

(*integer*) - the phone number being dialed (*text*) - name and IP address of the remote party (*name*) - CODEC used for the audio connection (*time*) - duration of the phone call

Notes

As some Voicetronix cards fail to detect loop drop correctly, with **loop-drop-detection** you can manage whether loop drop detection feature is enabled. The effect of not working loop-drop detection is call terminated at once when connection is established.

Some tips for testing balance registers:

- test is sensitive to noise from the phone, so it's recommended to cover mouth piece during it;
- **find-best-balance** can be interrupted by **clear-call** command;
- once best **balance-registers** value is known, it can be set manually to this best value for all voicetronix voice ports, which will use the same telephone line.

LineJack Voice Ports

Home menu level: ***/ip telephony voice-port linejack***

Property Description

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

name (*name*) - name given by the user or the default one

autodial (*integer*; default: "") - phone number which will be dialed immediately after the handset has been lifted. If this number is incomplete, then the remaining part has to be dialed on the dial-pad. If the number is incorrect, the line is hung up (FXO "line" port) or busy tone is played (FXS "phone" port). If the number is correct, then the appropriate number is dialed. If it is an incomming call from the PSTN line, then the direct-call mode is used - the line is picked up only after the remote party answers the call

playback-volume (*integer*: -48..48; default: **0**) - playback volume in dB

- **0** - 0dB meand no change to signal level

record-volume (*integer*: -48..48; default: **0**) - record volume in dB

- **0** - 0dB meand no change to signal level

ring-cadence (*text*) - a 16-symbol ring cadence for the phone, each symbol lasts 0.5 seconds, + means ringing, - means no ringing

region (*name*; default: **us**) - regional setting for the voice port. This setting is used for setting the parameters of PSTN line, as well as for detecting and generating the tones

aec (yes | no) - whether echo detection and cancellation is enabled

aec-tail-length (*short* | *medium* | *long*; default: **short**) - size of the buffer of echo detection

aec-nlp-threshold (*off* | *low* | *medium* | *high*; default: **low**) - level of cancellation of silent sounds

aec-attenuation-scaling (*integer*: 0..10; default: **4**) - factor of additional echo attenuation

aec-attenuation-boost (*integer*: 0..90; default: **0**) - level of additional echo attenuation

software-aec (yes | no) - software echo canceller (experimental, for most of the cards)

agc-on-playback (yes | no; default: **no**) - automatic gain control on playback (can not be used together with hardware voice codecs)

agc-on-record (yes | no; default: **no**) - automatic gain control on record (can not be used together with hardware voice codecs)

detect-cpt (yes | no; default: **no**) - automatically detect call progress tones

Command Description

blink - blink the LEDs of the specified voice port for five seconds after it is invoked. This command can be used to locate the respective card from several linejack cards (*name*) - card name to blink the LED of

clear-call - terminate a current call established with the specified voice port (*name*) - port name to clear call with

show-stats - show voice port statistics (*name*) - port name show statistics of (*time*) - maximal time of packet round trip (*integer*) - number of packets sent by this card (these packets are digitalized input of the voice port) (*integer*) - number of bytes sent by this card (these packets are digitalized input of the voice port) (*text*) - minimal/average/maximal intervals between packets sent (*integer*) - number of packets received by this card (these packets form analog output of the voice port) (*integer*) - number of bytes received by this card (these packets form analog output of the voice port) (*text*) - minimal/average/maximal intervals between packets received (*time*) - approximate delay time from the moment of receiving an audio packet from the IP network till it is played back over the telephony voice port. The value shown is never less than 30ms, although the actual delay time could be less. If the shown value is >40ms, then it is close (+/-1ms) to the actual delay time.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

monitor - monitor status of the voice port (*name*) - port name to monitor (*on-hook* | *off-hook* | *ring* | *connection* | *busy*) - current state of the port:

- **on-hook** - the handset is on-hook, no activity
- **off-hook** - the handset is off-hook, the number is being dialed
- **ring** - call in progress, direction of the call is shown by the direction property
- **connection** - the connection has been established
- **busy** - the connection has been terminated, the handset is still off-hook

(*phone* | *line*) - the active port of the card

- **phone** - telephone connected to the card (POTS FXS port)
- **line** - line connected to the card (PSTN FXO port)

(*ip-to-port* | *port-to-ip*) - direction of the call

- **ip-to-port** - call from the IP network to the voice card
- **port-to-ip** - call from the voice card to an IP address

(*plugged* | *unplugged*) - state of the PSTN line

- **plugged** - the telephone line is connected to the PSTN port of the card
- **unplugged** - there is no working line connected to the PSTN port of the card

(*integer*) - the phone number being dialed (*text*) - name and IP address of the remote party (*name*) - CODEC used for the audio connection (*time*) - duration of the phone call

Notes

When telephone line is connected to the 'line' port, green LED next to the port should be lit in some seconds. If telephone line disappear, the LED next to the 'line' port will change its state to red in an hour or when the line is activated (i.e. when somebody calls to/from it). When telephone line is plugged in the 'phone' port before the router is turned on, red LED next to the port will be lit. WARNING: do not plug telephone line into the 'phone' port when the router is running and green LED next to the port is lit - this might damage the card. The status of the 'phone' port is only detected on system startup.

PhoneJack Voice Ports

Home menu level: ***/ip telephony voice-port phonejack***

Property Description

name (*name*) - name given by the user or the default one

type (*read-only: phonejack* | *phonejack-lite* | *phonejack-pci*) - type of the card

autodial (*integer*; default: **""**) - phone number which will be dialed immediately after the handset has been lifted. If this number is incomplete, then the remaining part has to be dialed on the dial-pad. If the number is incorrect, busy tone is played. If the number is correct, then the appropriate number is dialed

playback-volume (*integer*: -48..48; default: **0**) - playback volume in dB

- **0** - 0dB meand no change to signal level

record-volume (*integer*: -48..48; default: **0**) - record volume in dB

- **0** - 0dB meand no change to signal level

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

ring-cadence (*text*) - a 16-symbol ring cadence for the phone, each symbol lasts 0.5 seconds, + means ringing, - means no ringing

region (*name*; default: **us**) - regional setting for the voice port. This setting is used for generating the dial tones

aec (yes | no) - whetheher echo detection and cancellation is enabled

aec-tail-length (*short* | *medium* | *long*; default: **short**) - size of the buffer of echo detection

aec-nlp-threshold (*off* | *low* | *medium* | *high*; default: **low**) - level of cancellation of silent sounds

aec-attenuation-scaling (*integer*: 0..10; default: **4**) - factor of additional echo attenuation

aec-attenuation-boost (*integer*: 0..90; default: **0**) - level of additional echo attenuation

software-aec (yes | no) - software echo canceller (experimental, for most of the cards)

agc-on-playback (yes | no; default: **no**) - automatic gain control on playback (can not be used together with hardware voice codecs)

agc-on-record (yes | no; default: **no**) - automatic gain control on record (can not be used together with hardware voice codecs)

detect-cpt (yes | no; default: **no**) - automatically detect call progress tones

Command Description

clear-call - terminate a current call established with the specified voice port (*name*) - port name to clear call with

show-stats - show voice port statistics (*name*) - port name show statistics of (*time*) - maximal time of packet round trip (*integer*) - number of packets sent by this card (these packets are digitalized input of the voice port) (*integer*) - number of bytes sent by this card (these packets are digitalized input of the voice port) (*text*) - minimal/average/maximal intervals between packets sent (*integer*) - number of packets received by this card (these packets form analog output of the voice port) (*integer*) - number of bytes received by this card (these packets form analog output of the voice port) (*text*) - minimal/average/maximal intervals between packets received (*time*) - approximate delay time from the moment of receiving an audio packet from the IP network till it is played back over the telephony voice port. The value shown is never less than 30ms, although the actual delay time could be less. If the shown value is >40ms, then it is close (+/-1ms) to the actual delay time.

monitor - monitor status of the voice port (*name*) - port name to monitor (*on-hook* | *off-hook* | *ring* | *connection* | *busy*) - current state of the port:

- **on-hook** - the handset is on-hook, no activity
- **off-hook** - the handset is off-hook, the number is being dialed
- **ring** - call in progress, direction of the call is shown by the direction property
- **connection** - the connection has been established
- **busy** - the connection has been terminated, the handset is still off-hook

(*phone* | *line*) - the active port of the card

- **phone** - telephone connected to the card (POTS FXS port)
- **line** - line connected to the card (PSTN FXO port)

(*ip-to-port* | *port-to-ip*) - direction of the call

- **ip-to-port** - call from the IP network to the voice card
- **port-to-ip** - call from the voice card to an IP address

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

(*plugged* | *unplugged*) - state of the PSTN line

- **plugged** - the telephone line is connected to the PSTN port of the card
- **unplugged** - there is no working line connected to the PSTN port of the card

(*integer*) - the phone number being dialed (*text*) - name and IP address of the remote party (*name*) - CODEC used for the audio connection (*time*) - duration of the phone call

Zaptel Voice Ports

Home menu level: ***/ip telephony voice-port zaptel***

Property Description

name (*name*) - name given by the user or the default one

autodial (*integer*; default: **""**) - phone number which will be dialed immediately after the handset has been lifted. If this number is incomplete, then the remaining part has to be dialed on the dial-pad. If the number is incorrect, the line is hung up. If the number is correct, then the appropriate number is dialed (the direct-call mode is used - the line is picked up only after the remote party answers the call)

playback-volume (*integer*: -48..48; default: **0**) - playback volume in dB

- **0** - 0dB meand no change to signal level

record-volume (*integer*: -48..48; default: **0**) - record volume in dB

- **0** - 0dB meand no change to signal level

region (*name*; default: **us**) - regional setting for the voice port. This setting is used for setting the parameters of PSTN line, as well as for detecting and generating the tones

aec (yes | no) - whetеher echo detection and cancellation is enabled

aec-tail-length (*short* | *medium* | *long*; default: **short**) - size of the buffer of echo detection

aec-nlp-threshold (*off* | *low* | *medium* | *high*; default: **low**) - level of cancellation of silent sounds

aec-attenuation-scaling (*integer*: 0..10; default: **4**) - factor of additional echo attenuation

aec-attenuation-boost (*integer*: 0..90; default: **0**) - level of additional echo attenuation

software-aec (yes | no) - software echo canceller (experimental, for most of the cards)

agc-on-playback (yes | no; default: **no**) - automatic gain control on playback (can not be used together with hardware voice codecs)

agc-on-record (yes | no; default: **no**) - automatic gain control on record (can not be used together with hardware voice codecs)

detect-cpt (yes | no; default: **no**) - automatically detect call progress tones

Command Description

clear-call - terminate a current call established with the specified voice port (*name*) - port name to clear call with

show-stats - show voice port statistics (*name*) - port name show statistics of (*time*) - maximal time of packet round trip (*integer*) - number of packets sent by this card (these packets are digitalized input of the voice port) (*integer*) - number of bytes sent by this card (these packets are digitalized input of the voice port) (*text*) - minimal/average/maximal intervals between packets sent (*integer*) -

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

number of packets received by this card (these packets form analog output of the voice port) (*integer*) - number of bytes received by this card (these packets form analog output of the voice port) (*text*) - minimal/average/maximal intervals between packets received (*time*) - approximate delay time from the moment of receiving an audio packet from the IP network till it is played back over the telephony voice port. The value shown is never less than 30ms, although the actual delay time could be less. If the shown value is >40ms, then it is close (+/-1ms) to the actual delay time.

monitor - monitor status of the voice port (*name*) - port name to monitor (*on-hook* | *off-hook* | *ring* | *connection* | *busy*) - current state of the port:

- **on-hook** - the handset is on-hook, no activity
- **off-hook** - the handset is off-hook, the number is being dialed
- **ring** - call in progress, direction of the call is shown by the direction property
- **connection** - the connection has been established
- **busy** - the connection has been terminated, the handset is still off-hook

(*ip-to-port* | *port-to-ip*) - direction of the call

- **ip-to-port** - call from the IP network to the voice card
- **port-to-ip** - call from the voice card to an IP address

(*plugged* | *unplugged*) - state of the PSTN line

- **plugged** - the telephone line is connected to the PSTN port of the card
- **unplugged** - there is no working line connected to the PSTN port of the card

(*integer*) - the phone number being dialed (*text*) - name and IP address of the remote party (*name*) - CODEC used for the audio connection (*time*) - duration of the phone call

ISDN Voice Ports

Home menu level: ***/ip telephony voice-port isdn***

Property Description

name (*name*) - name given by the user or the default one

msn (*integer*) - telephone number of the ISDN voice port (ISDN MSN number)

lmsn (*text*) - msn pattern to listen on. It determines which calls from the ISDN line this voice port should answer. If left empty, msn is used

autodial (*integer*; default: **""**) - phone number which will be dialed immediately on each incoming ISDN call. If this number contains 'm', then it will be replaced by originally called (ISDN) telephone number. If this number is incomplete, then the remaining part has to be dialed by the caller. If the number is incorrect, call is refused. If the number is correct, then the appropriate number is dialed. For that direct-call mode is used - the line is picked up only after the remote party answers the call

playback-volume (*integer*: -48..48; default: **0**) - playback volume in dB

- **0** - 0dB meand no change to signal level

record-volume (*integer*: -48..48; default: **0**) - record volume in dB

- **0** - 0dB meand no change to signal level

region (*name*; default: **us**) - regional setting for the voice port. This setting is used for setting the parameters of PSTN line, as well as for detecting and generating the tones

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

aec (yes | no) - whetheher echo detection and cancellation is enabled

aec-tail-length (*short* | *medium* | *long*; default: **short**) - size of the buffer of echo detection

software-aec (yes | no) - software echo canceller (experimental, for most of the cards)

agc-on-playback (yes | no; default: **no**) - automatic gain control on playback (can not be used together with hardware voice codecs)

agc-on-record (yes | no; default: **no**) - automatic gain control on record (can not be used together with hardware voice codecs)

Command Description

clear-call - terminate a current call established with the specified voice port (*name*) - port name to clear call with

show-stats - show voice port statistics (*name*) - port name show statistics of (*time*) - maximal time of packet round trip (*integer*) - number of packets sent by this card (these packets are input of the voice port) (*integer*) - number of bytes sent by this card (these packets are input of the voice port) (*text*) - minimal/average/maximal intervals between packets sent (*integer*) - number of packets received by this card (these packets form output of the voice port) (*integer*) - number of bytes received by this card (these packets form output of the voice port) (*text*) - minimal/average/maximal intervals between packets received (*time*) - approximate delay time from the moment of receiving an audio packet from the IP network till it is played back over the telephony voice port. The value shown is never less than 30ms, although the actual delay time could be less. If the shown value is >40ms, then it is close (+/-1ms) to the actual delay time.

monitor - monitor status of the voice port (*name*) - port name to monitor (*on-hook* | *off-hook* | *ring* | *connection* | *busy*) - current state of the port:

- **on-hook** - the handset is on-hook, no activity
- **off-hook** - the handset is off-hook, the number is being dialed
- **ring** - call in progress, direction of the call is shown by the direction property
- **connection** - the connection has been established
- **busy** - the connection has been terminated, the handset is still off-hook

(*ip-to-port* | *port-to-ip*) - direction of the call

- **ip-to-port** - call from the IP network to the voice card
- **port-to-ip** - call from the voice card to an IP address

(*integer*) - the phone number being dialed (*text*) - name and IP address of the remote party (*name*) - CODEC used for the audio connection (*time*) - duration of the phone call

Notes

In contrary to analog voice ports phonejack, linejack, voicetronix, zaptel), which are as many as the number of cards installed, the isdn ports can be added as many as desired.

- **;** - separates pattern entries (more than one pattern can be specified this way)
- **?** - matches one character
- ***** - matches zero or more characters
- **[ ]** - matches any single character from the set in brackets
- **[^]** - matches any single character not from the set in brackets

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

There is a possibility to enter some special symbols in **lmsn** property. Meaning of the special symbols:

Voice Port for Voice over IP (voip)

Home menu level: ***/ip telephony voice-port voip***

Description

The voip voice ports are virtual ports, which designate a voip channel to another host over the IP network. You must have at least one voip voice port to be able to make calls to other H.323 devices over IP network.

Property Description

name (*name*) - name given by the user or the default one

remote-address (*IP address*; default: **0.0.0.0**) - IP address of the remote party (IP telephone or gateway) associated with this voice port. If the call has to be performed through this voice port, then the specified IP address is called. If there is an incoming call from the specified IP address, then the parameters of this voice port are used. If there is an incoming call from an IP address, which is not specified in any of the voip voice port records, then the default record is used. If there is no default record, then default values are used

- **0.0.0.0** - the record with this IP address will specify the default values for an incomming call

autodial (*integer*) - phone number which will be added in front of the telephone number received over the IP network. In most cases it should be blank

jitter-buffer (*time*: 0..1000ms; default: **100ms**) - size of the jitter buffer

- **0** - the size of it is adjusted automatically during the conversation, to keep amount of lost packets under 1%

silence-detection (yes | no; default: **no**) - whether silence is detected and no audio data is sent over the IP network during the silence period

prefered-codec (*name*; default: **none**) - the preferred codec to be used for this voip voice port. If possible, the specified codec will be used

- **none** - there is no preferred codec defined for this port, so whichever codec advised by the remote peer will be used (if it is supported)

fast-start (yes | no; default: **yes**) - allow or disallow the fast start. The fast start allows establishing the audio connection in a shorter time. However, not all H.323 endpoints support this feature. Therefore, it should be turned off, if there are problems to establish telephony connection using the fast start mode

Numbers

Description

This is the so-called "routing table" for voice calls. This table assigns numbers to the voice ports.The main function of the numbers routing table is to determine:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- to which voice port route the call
- what number to send over to the remote party

Property Description

dst-pattern (*integer*) - pattern of the telephone number. Symbol '.' designate any digit, symbol '_' (only as the last one) designate any symbols (i.e. any number of characters can follow, ended with '#' button)

voice-port (*name*) - voice port to be used when calling the specified telephone number

prefix (*integer*) - prefix, which will be used to substitute the known part of the dst-pattern, i.e., the part containing digits. The dst-pattern argument is used to determine which voice port to be used, whereas the prefix argument designates the number to dial over the voice port (be sent over to the remote party). If the remote party is an IP telephony gateway, then the number will be used for making the call

Notes

More than one entry can be added with exactly the same **dst-pattern**. If first one of them is already busy, next one with the same **dst-pattern** is used. Telephony number entries can be moved, to select desired order.

Example

Let us consider the following example for the number table:

```
[admin@MikroTik] ip telephony numbers> print
Flags: I - invalid, X - disabled, D - dynamic, R - registered
  #     DST-PATTERN                    VOICE-PORT PREFIX
  0     12345                          XX
  1     1111.                          YY
  2     22...                          ZZ         333
  3     ...                            QQ         55

[admin@MikroTik] ip telephony numbers>
```

We will analyze the Number Received (nr) - number dialed at the telephone, or received over the line, the Voice Port (vp) - voice port to be used for the call, and the Number to Call (nc) - number to be called over the Voice Port.

- If nr=55555, it does not match any of the destination patterns, therefore it is rejected
- If nr=123456, it does not match any of the destination patterns, therefore it is rejected
- If nr=1234, it does not match any of the destination patterns (incomplete for record #0), therefore it is rejected
- If nr=12345, it matches the record #0, therefore number "" is dialed over the voice port XX
- If nr=11111, it matches the record #1, therefore number "1" is dialed over the voice port YY
- If nr=22987, it matches the record #2, therefore number "333987" is dialed over the voice port ZZ
- If nr=22000, it matches the record #2, therefore number "333000" is dialed over the voice port

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

ZZ

- If nr=444, it matches the record #3, therefore number "55444" is dialed over the voice port QQ

Let us add a few more records:

```
[admin@MikroTik] ip telephony numbers> print
Flags: I - invalid, X - disabled, D - dynamic, R - registered
  #     DST-PATTERN                    VOICE-PORT PREFIX
  0     12345                          XX
  1     1111.                          YY
  2     22...                          ZZ         333
  3     ...                            QQ         55
  4     222                            KK         44444
  5     3..                            LL         553

[admin@MikroTik] ip telephony numbers>
```

- If nr=222 => the best match is the record #4 => nc=44444, vp=KK (note: the 'best match' means that it has the most coinciding digits between the nr and destination pattern).
- If nr=221 => incomplete record #2 => call is rejected
- If nr=321 => the best match is the record #5 => nc=55321, vp=LL
- If nr=421 => matches the record #3 => nc=55421, vp=QQ
- If nr=335 => the best match is the record #5 => nc=55321, vp=LL

Let us add a few more records:

```
[admin@MikroTik] ip telephony numbers> print
Flags: I - invalid, X - disabled, D - dynamic, R - registered
Flags: I - invalid, X - disabled, D - dynamic, R - registered
  #     DST-PATTERN                    VOICE-PORT PREFIX
  0     12345                          XX
  1     1111.                          YY
  2     22...                          ZZ         333
  3     ...                            QQ         55
  4     222                            KK         44444
  5     3..                            LL         553
  6     33...                          MM         33
  7     11.                            NN         7711

[admin@MikroTik] ip telephony numbers>
```

- If nr=335 => incomplete record #6 => the call is rejected. The nr=335 fits perfectly both the record #3 and #5. The #5 is chosen as the 'best match' candidate at the moment. Furthermore, there is record #6, which has two matching digits (more than for #3 or #5). Therefore the #6 is chosen as the 'best match'. However, the record #6 requires five digits, but the nr has only three. Two digits are missing, therefore the number is incomplete. Two additional digits would be needed to be entered on the dialpad. If the number is sent over from the network, it is rejected.
- If nr=325 => matches the record #5 => nc=55325, vp=LL
- If nr=33123 => matches the record #6 => nc=33123, vp=MM
- If nr=123 => incomplete record #0 => call is rejected
- If nr=111 => incomplete record #1 => call is rejected

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- If nr=112 => matches the record #7 => nc=77112, vp=NN
- If nr=121 => matches the record #3 => nc=55121, vp=QQ

It is impossible to add the following records:

```
#     DST-PATTERN                   VOICE-PORT PREFIX
                                                  reason:
      11                            DD            conflict with record # 1
                                                  and # 7
      11..                          DD            conflict with record # 7
      111                           DD            conflict with record # 1
      22.                           DD            conflict with record # 2
      .....                         DD            conflict with record # 3
```

Regional Settings

Home menu level: ***/ip telephony region***

Description

Regional settings are used to adjust the voice port properties to the PSTN system or the PBX. For example, to detect hang-up from line, there has to be correct regional setting (correct busy-tone-frequency and busy-tone-cadence). Without that, detect-cpt parameter the voice port has to be enabled.

Property Description

name (*name*) - name of the regional setting

busy-tone-cadence (*integer*: 0..30000; default: **500,500**) - busy tone cadence in ms

- **0** - end of cadence

busy-tone-frequency (*integer*: 20..2000 | *integer*: -24..6; default: **440x0**) - frequency and volume gain of busy tone, Hz x dB

data-access-arrangement (*australia* | *france* | *germany* | *japan* | *uk* | *us*; default: **us**) - ring voltage, impedance setting for line-jack card

dial-tone-frequency (*integer*: 20..2000 | *integer*: -24..6; default: **440x0**) - frequency and volume gain of dial tone, Hz x dB

dtmf-tone-cadence (*integer*: 0..30000; default: **180,60**) - Dual Tone Multi Frequency tone cadence in ms

- **0** - end of cadence

dtmf-tone-volume (*integer*: -24..6; default: **-3,-3**) - Dual Tone Multi Frequency tone volume in dB

ring-tone-cadence (*integer*: 0..30000; default: **1000,2000**) - Ring tone cadence in ms

- **0** - end of cadence

ring-tone-frequency (*integer*: 20..2000 | *integer*: -24..6; default: **440x0**) - frequency and volume gain of busy tone, Hz x dB

Notes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

To generate a tone, frequency and cadence arguments are used. The dialtone always is continuous signal, therefore it does not have the cadence argument. In order to detect dialtone, it should be at least 100ms long.

There are 10 pre-defined regions, which can not be deleted (but may be changed)

Audio CODECs

Home menu level: ***/ip telephony codec***

Description

CODECs are listed according to their priority of use. The highest priority is at the top. CODECs can be enabled, disabled and moved within the list. When connecting with other H.323 systems, the protocol will negotiate the CODEC which both of them support according to the priority order.

The hardware codecs (/hw) are built-in CODECs supported by some cards.

The choice of the CODEC type is based on the throughput and speed of the network. Better audio quality can be achieved by using CODEC requiring higher network throughput. The highest audio quality can be achieved by using the G.711-uLaw CODEC requiring 64kb/s throughput for each direction of the call. It is used mostly within a LAN. The G.723.1 CODEC is the most popular one to be used for audio connections over the Internet. It requires only 6.3kb/s throughput for each direction of the call.

Example

```
[admin@MikroTik] ip telephony codec> print
Flags: X - disabled
  #   NAME
  0   G.723.1-6.3k/sw
  1   G.728-16k/hw
  2   G.711-ALaw-64k/hw
  3   G.711-uLaw-64k/hw
  4   G.711-uLaw-64k/sw
  5   G.711-ALaw-64k/sw
  6   G.729A-8k/sw
  7   GSM-06.10-13.2k/sw
  8   LPC-10-2.5k/sw
  9   G.723.1-6.3k/hw
 10   G.729-8k/sw
[admin@MikroTik] ip telephony codec>
```

AAA

Home menu level: ***/ip telephony aaa***

Description

AAA (Authentication Authorization Accounting) can be used to configure the RADIUS accounting feature.

- **NAS-Identifier** - router name (from /system identity print)
- **NAS-IP-Address** - router's local IP address which the connection was established to (if exist)

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **NAS-Port-Type** - always Async
- **Event-Timestamp** - data and time of the event
- **Acct-Session-Time** - current connection duration (only in INTERIM-UPDATE and STOP records)
- **Acct-Output-Packets** - sent RTP (Real-Time Transport Protocol) packet count (only in INTERIM-UPDATE and STOP records)
- **Acct-Output-Packets** - sent RTP (Real-Time Transport Protocol) packet count (only in INTERIM-UPDATE and STOP records)
- **Acct-Input-Packets** - received RTP (Real-Time Transport Protocol) packet count (only in INTERIM-UPDATE and STOP records)
- **Acct-Output-Octets** - sent byte count (only in INTERIM-UPDATE and STOP records)
- **Acct-Input-Octets** - received byte count (only in INTERIM-UPDATE and STOP records)
- **Acct-Session-Id** - unique session participient ID
- **h323-disconnect-cause** - session disconnect reason (only in STOP records):
- **h323-disconnect-time** - session disconnect time (only in INTERIM-UPDATE and STOP records)
- **h323-connect-time** - session establish time (only in INTERIM-UPDATE and STOP records)
- **h323-gw-id** - name of gateway emitting message (should be equal to NAS-Identifier)
- **h323-call-type** - call leg type (should be VoIP)
- **h323-call-origin** - indicates origin of call relatively to the gateway (answer for calls from IP network, originate - to IP network)
- **h323-setup-time** - call setup time
- **h323-conf-id** - unique session ID
- **h323-remote-address** - the remote address of the session
- **NAS-Port-Id** - voice port ID
- **Acct-Status-Type** - record type (START when session is established; STOP when session is closed; INTERIM-UPDATE (ALIVE)session is alive). The time between the interim-update messages is defined by the interim-update-interval parameter (if it is set to 0s, there will be no such messages)

The contents of the CDR (Call Detail Record) are as follows:

- - **0** - Local endpoint application cleared call
  - **1** - Local endpoint did not accept call
  - **2** - Local endpoint declined to answer call
  - **3** - Remote endpoint application cleared call
  - **4** - Remote endpoint refused call
  - **5** - Remote endpoint did not answer in required time
  - **6** - Remote endpoint stopped calling
  - **7** - Transport error cleared call
  - **8** - Transport connection failed to establish call
  - **9** - Gatekeeper has cleared call

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **10** - Call failed as could not find user (in GK)
- **11** - Call failed as could not get enough bandwidth
- **12** - Could not find common capabilities
- **13** - Call was forwarded using FACILITY message
- **14** - Call failed a security check and was ended
- **15** - Local endpoint busy
- **16** - Local endpoint congested
- **17** - Remote endpoint busy
- **18** - Remote endpoint congested
- **19** - Could not reach the remote party
- **20** - The remote party is not running an endpoint
- **21** - The remote party host off line
- **22** - The remote failed temporarily app may retry

Property Description

use-radius-accounting (yes | no; default: **no**) - whether to use radius accounting or not

interim-update (*integer*; default: **0**) - defines time interval between communications with the router. If this time will exceed, RADIUS server will assume that this connection is down. This value is suggested not to be less than 3 minutes

- **0** - no interim-update messages are sent at all

Notes

All the parameters, which names begin with **h323**, are CISCO vendor specific Radius attributes

Gatekeeper

Home menu level: ***/ip telephony gatekeeper***

Description

For each H.323 endpoint gatekeeper stores its telephone numbers. So, gatekeeper knows all telephone numbers for all registered endpoints. And it knows which telephone number is handled by which endpoint. Mapping between endpoints and their telephone numbers is the main functionality of gatekeepers.

If endpoint is registered to endpoint, it does not have to know every single endpoint and every single telephone number, which can be called. Instead, every time some number is dialed, endpoint asks gatekeeper for destination endpoint to call by providing called telephone number to it.

MikroTik IP telephony package includes a very simple gatekeeper. This gatekeeper can be activated by setting **gatekeeper** parameter to **local**. In this case the local endpoint automatically is registered to the local gatekeeper. And any other endpoint can register to this gatekeeper too.

Registered endpoints are added to the **/ip telephony voice-port voip** table. Those entries are

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

marked as dynamic and can not be removed or changed. If there already was an voip entry with the same IP address, it is marked as registred. Remote-address can not be changed for these entries too, but registered voip voice ports can be removed - they will stay as dynamic ones. If there already is a dynamic voip voice port and a static one with the same IP address is added, then instead of dynamic entry, registered will appear.

Dynamic entries disappear when corresponding endpoint unregisters itself from the gatekeeper. Registered entries are static and will stay even after that endpoint will be unregistered from this gatekeeper.

Registered telephone numbers are added to **/ip telephony numbers** table. Here is exactly the same idea behind dynamic and registered telephone numbers as it is with voip voice ports.

When an endpoint registers to the gatekeeper, it sends its own telephone numbers (aliases and prefixes) within this registration request. **/ip telephony numbers** entry is registered to the endpoint only if voice-port for that entry is local (not voip). If **dst-pattern** contains '.' or '_', it is sent as prefix, otherwise - as alias. The known part of the **dst-pattern** is sent as prefix. If there is no known part (**dst-pattern** is "_" or "...", for example), then this entry is not sent at all.

Property Description

gatekeeper (*none* | *local* | *remote*; default: **none**) - Gatekeeper type to use

- **none** - don't use any gatekeeper at all
- **local** - start and use local gatekeeper
- **remote** - use some other gatekeeper

remote-address (*IP address*; default: **0.0.0.0**) - IP address of remote gatekeeper to use. If set to 0.0.0.0, broadcast gatekeeper discovery is used

remote-id (*name*) - name of remote gatekeeper to use. If left empty, first available gatekeeper will be used. Name of locally started gatekeeper is the same as system identity

registered (*read-only:* yes | no) - shows whether local H.323 endpoint is registered to any gatekeeper

registered-with (*read-only: name*) - name of gatekeeper to which local H.323 endpoint is registered

Example

In most simple case with one phonejack card and some remote gatekeeper, configuration can be as follows:

```
[admin@MikroTik] ip telephony voice-port> print
Flags: X - disabled
  #   NAME                        TYPE            AUTODIAL
  0   phonejack1                phonejack
  1   voip1                       voip

[admin@MikroTik] ip telephony voice-port voip> print
Flags: X - disabled, D - dynamic, R - registered
  #    NAME      AUTODIAL REMOTE-ADDRESS  JITTER-BUFFER PREFERED-CODEC  SIL FAS
  0    voip1              0.0.0.0         0s            none            no  yes

[admin@MikroTik] ip telephony numbers> print
Flags: I - invalid, X - disabled, D - dynamic, R - registered
  #     DST-PATTERN             VOICE-PORT              PREFIX
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  0     11                      phonejack1
  1     _                       voip1

[admin@MikroTik] ip telephony gatekeeper> print
         gatekeeper: remote
          remote-id: ""
     remote-address: 10.0.0.98
         registered: yes
    registered-with: "MikroTik@10.0.0.98"
```

In this case this endpoint will register to gatkeeper with the IP address of 10.0.0.98 and telephone number 11. Every call to telephone number 11 will be transfered from gatekeeper to this endpoint. And this endpoint will route this call to phonejack1 voice port. On any other telephone number gatekeeper will be asked for real destination. From this endpoint it will be possible to call all the endpoints, which are registered to the same gatekeeper. If that gatekeeper has static entries about endpoints, which are not registered to gatekeeper, it still will be possible to call those endpoints by those statically defined telephone numbers at gatekeeper.

Example

For example, if numbers table is like this:

```
[admin@MikroTik] ip telephony numbers> print
Flags: I - invalid, X - disabled, D - dynamic, R - registered
  #     DST-PATTERN             VOICE-PORT              PREFIX
  0     1.                      phonejack1
  1     128                     voip1                   128
  2     78                      voip2                   78
  3     77                      phonejack1
  4     76                      phonejack1              55
  5     _                       voip1
```

then entries 0, 3 and 4 will be sent to the gatekeeper, others are voip voice ports and are ignored. Entry 0 will be sent as prefix 1, entry 3 - as alias 77, and entry 4 - as alias 76.

If IP address of local endpoint is 10.0.0.100, then gatekeeper voip and numbers tables will look as follows:

```
      [admin@MikroTik] ip telephony voice-port voip> print
Flags: X - disabled, D - dynamic, R - registered
  #    NAME      AUTODIAL REMOTE-ADDRESS  JITTER-BUFFER PREFERED-CODEC  SIL FAS
  0    tst-2.5            10.0.0.101      0s            none            no  yes
  1  D local              127.0.0.1       100ms         none            no  yes
  2  D 10.0.0...          10.0.0.100      100ms         none            no  yes

[admin@MikroTik] ip telephony numbers> print
Flags: I - invalid, X - disabled, D - dynamic, R - registered
  #     DST-PATTERN             VOICE-PORT              PREFIX
  0     78                      linejack1
  1     3...                    vctx1
  2     33_                     voip1
  3     5..                     voip1
  4  XD 78                      local                   78
  5  XD 3_                      local                   3
  6   D 76                      10.0.0.100              76
  7   D 77                      10.0.0.100              77
  8   D 1_                      10.0.0.100              1
```

Here we can see how aliases and prefixes are added to numbers table. Entries 0..3 are static. Entries 4 and 5 are added by registering the local endpoint to the local gatekeeper. Entries 6..8 are added by

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

registering endpoint (with IP address 10.0.0.100) to the local gatekeeper.

For prefixes, '_' is added at the end of dst-pattern to allow any additional digits to be added at the end.

Local endpoint is registered to the local gatekeeper too. So, local aliases and prefixes are added as dynamic numbers too. Only, as they are local and corresponding number entries already exist in the number table, then these dynamically added entries are disabled by default.

If any registered telephone number will conflict with some existing telephone numbers entry, it will be added as disabled and dynamic.

If in gatekeeper's numbers table there already exists exactly the same dst-pattern as some other endpoint is trying to register, this gatekeeper registration for that endpoint will fail.

Troubleshooting

Description

- **The IP Telephony does not work after upgrading from 2.5.x version** - You need to completely reinstall the router using any installation procedure. You may keep the configuration using either the installation program option or the backup file.
- **The IP Telephony gateway does not detect the drop of the line when connected to some PBXs** - Different regional setting should be used to match the parameters of the PBX. For example, try using uk for Meridian PBX.
- **The IP Telephone does not call the gateway, but gives busy signal** - Enable the logging of IP telephony events under /system logging facility. Use the monitoring function for voice ports to debug your setup while making calls.
- **The IP telephony is working without NAT, but sound goes only in one direction** - Disable H323 service port in firewall: /ip firewall service-port set h323 disabled=yes
- **The IP Telephony does not work through NAT** - Enable H323 service port in firewall: /ip firewall service-port set h323 disabled=no

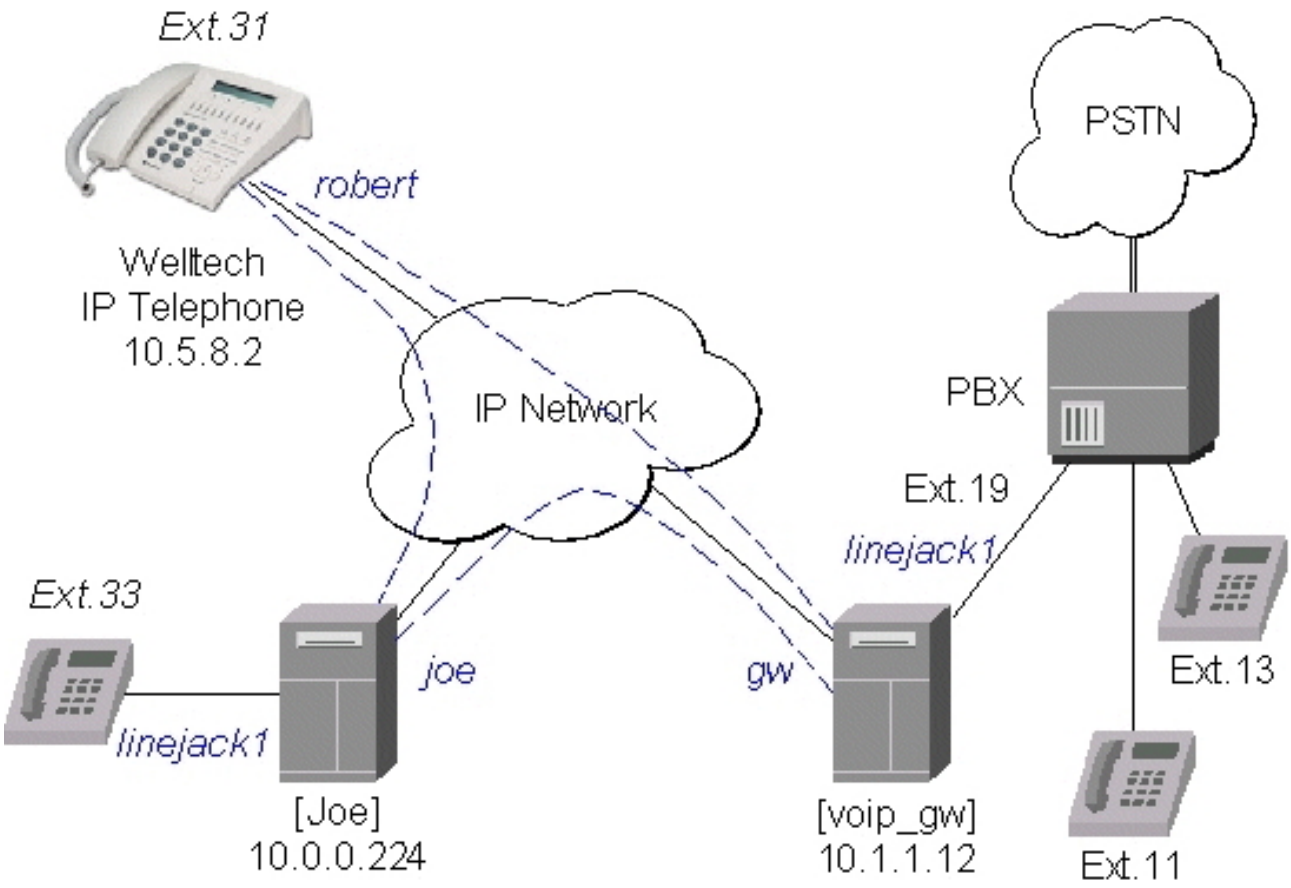
A simple example

Description

The following describes examples of some useful IP telephony applications using MikroTik RouterOS.

Let us consider the following example of IP telephony gateway, one MikroTik IP telephone, and one Welltech LAN Phone 101 setup:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



Setting up the MikroTik IP Telephone

If you pick up the handset, a dialtone should be heard.

The basic telephony configuration should be as follows:

- Add a voip voice port to the **/ip telephony voice-port voip** for each of the devices you want to call, or want to receive calls from, i.e., (the IP telephony gateway 10.1.1.12 and the Welltech IP telephone 10.5.8.2):

```
[admin@Joe] ip telephony voice-port voip> add name=gw remote-address=10.1.1.12
[admin@Joe] ip telephony voice-port voip> add name=rob remote-address=10.5.8.2
[admin@Joe] ip telephony voice-port voip> print Flags: X - disabled, D - dynamic, R
- registered # NAME AUTODIAL REMOTE-ADDRESS JITTER-BUFFER PREFERED-CODEC SIL FAS 0
gw 10.1.1.12 100ms none no yes 1 rob 10.5.8.2 100ms none no yes [admin@Joe] ip
telephony voice-port voip>
```

  You should have three vioce ports now:

```
[admin@Joe] ip telephony voice-port> print Flags: X - disabled # NAME TYPE AUTODIAL
0 linejack1 linejack 1 gw voip 2 rob voip [admin@Joe] ip telephony voice-port>
```

- Add at least one unique number to the **/ip telephony numbers** for each voice port. This number will be used to call that port:

```
[admin@Joe] ip telephony numbers> add dst-pattern=31 voice-port=rob [admin@Joe] ip
telephony numbers> add dst-pattern=33 voice-port=linejack1 [admin@Joe] ip telephony
numbers> add dst-pattern=1. voice-port=gw prefix=1 [admin@Joe] ip telephony numbers>
print Flags: I - invalid, X - disabled, D - dynamic, R - registered # DST-PATTERN
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
VOICE-PORT PREFIX 0 31 rob 31 1 33 linejack1 2 1. gw 1 [admin@Joe] ip telephony
numbers>
```

Here, the **dst-pattern=31** is to call the Welltech IP Telephone, if the number 31 is dialed on the dialpad. The **dst-pattern=33** is to ring the local telephone, if a call for number 33 is received over the network. Anything starting with digit '1' would be sent over to the IP Telephony gateway.

Making calls from the IP telephone 10.0.0.224:

- To call the IP telephone 10.5.8.2, it is enough to lift the handset and dial the number 31
- To call the PBX extension 13, it is enough to lift the handset and dial the number 13
  After establishing the connection with 13, the voice port monitor shows:

```
[admin@Joe] ip telephony voice-port linejack> monitor linejack status: connection
port: phone direction: port-to-ip line-status: unplugged phone-number: 13
remote-party-name: PBX_Line [10.1.1.12] codec: G.723.1-6.3k/hw duration: 16s
[admin@Joe] ip telephony voice-port linejack>
```

Setting up the IP Telephony Gateway

The IP telephony gateway [voip_gw] requires the following configuration:

- Set the regional setting to match our PBX. The **mikrotik** region will be used in thisn example:

```
[admin@voip_gw] ip telephony voice-port linejack> set linejack1 region=mikrotik
[admin@voip_gw] ip telephony voice-port linejack> print
Flags: X - disabled
  0   name="linejack1" autodial="" region=mikrotik playback-volume=0
      record-volume=0 ring-cadence="++-++--- ++-++---" agc-on-playback=no
      agc-on-record=no aec=yes aec-tail-length=short aec-nlp-threshold=low
      aec-attenuation-scaling=4 aec-attenuation-boost=0 software-aec=no
      detect-cpt=yes

[admin@voip_gw] ip telephony voice-port linejack>
```

- Add a voip voice port to the **/ip telephony voice-port voip** for each of the devices you want to call, or want to receive calls from, i.e., (the IP telephone 10.0.0.224 and the Welltech IP telephone 10.5.8.2):

```
[admin@voip_gw] ip telephony voice-port voip> add name=joe \
\... remote-address=10.0.0.224
[admin@voip_gw] ip telephony voice-port voip> add name=rob \
\... remote-address=10.5.8.2 prefered-codec=G.723.1-6.3k/hw
[admin@voip_gw] ip telephony voice-port voip> print
Flags: X - disabled, D - dynamic, R - registered
  #    NAME      AUTODIAL REMOTE-ADDRESS  JITTER-BUFFER PREFERED-CODEC  SIL FAS
  0    joe               10.0.0.224      100ms         none            no yes
  1    rob               10.5.8.2        100ms         G.723.1-6.3k/hw no yes
[admin@voip_gw] ip telephony voice-port voip>
```

- Add number records to the **/ip telephony numbers**, so you are able to make calls:

```
[admin@voip_gw] ip telephony numbers> add dst-pattern=31 voice-port=rob prefix=31
[admin@voip_gw] ip telephony numbers> add dst-pattern=33 voice-port=joe prefix=33
[admin@voip_gw] ip telephony numbers> add dst-pattern=1. voice-port=linejack1 \
\... prefix=1
[admin@voip_gw] ip telephony numbers> print
Flags: I - invalid, X - disabled, D - dynamic, R - registered
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
  #     DST-PATTERN             VOICE-PORT              PREFIX
  0     31                          rob                 31
  1     33                          joe                 33
  2     1.                          linejack1           1
[admin@voip_gw] ip telephony numbers>
```

Making calls through the IP telephony gateway:

- To dial the IP telephone 10.0.0.224 from the office PBX line, the extension number 19 should be dialed, and, after the dial tone has been received, the number 33 should be entered. Thus, the telephone [Joe] is ringed.
  After establishing the voice connection with '33' (the call has been answered), the voice port monitor shows:
  ```
  [admin@voip_gw] ip telephony voice-port linejack> monitor linejack1 status:
  connection port: line direction: port-to-ip line-status: plugged phone-number: 33
  remote-party-name: linejack1 [10.0.0.224] codec: G.723.1-6.3k/hw duration: 1m46s
  [admin@voip_gw] ip telephony voice-port linejack>
  ```
- To dial the IP telephone 10.5.8.2 from the office PBX line, the extension number 19 should be dialed, and, after the dial tone has been received, the number 31 should be entered.

Setting up the Welltech IP Telephone

Please follow the documentation from ***www.welltech.com.tw*** on how to set up the Welltech LAN Phone 101. Here we give just brief recommendations:

1. We recommend to upgrade the Welltech LAN Phone 101 with the latest application software. Telnet to the phone and check what you have, for example:

```
usr/config$ rom -print

Download Method  :  TFTP
 Server Address  :  10.5.8.1

   Hardware Ver. :  4.0
       Boot Rom  :  nblp-boot.102a
Application Rom  :  wtlp.108h
        DSP App  :  48302ce3.127
     DSP Kernel  :  48302ck.127
  DSP Test Code  :  483cbit.bin
  Ringback Tone  :  wg-ringbacktone.100
      Hold Tone  :  wg-holdtone10s.100
  Ringing Tone1  :  ringlow.bin
  Ringing Tone2  :  ringmid.bin
  Ringing Tone3  :  ringhi.bin

usr/config$
```

2. Check if you have the codecs arranged in the desired order:

```
usr/config$ voice -print
Voice codec setting relate information
    Sending packet size  :
            G.723.1      : 30 ms
            G.711A       : 20 ms
            G.711U       : 20 ms
            G.729A       : 20 ms
            G.729        : 20 ms
    Priority order codec :
            g7231 g711a g711u g729a g729
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
    Volume levels        :
            voice volume : 54
            input gain   : 26
             dtmf volume : 23
Silence suppression & CNG:
           G.723.1       : Off
    Echo canceller       : On
 JitterBuffer Min Delay  : 90
 JitterBuffer Max Delay  : 150
usr/config$
```

3. Make sure you have set the H.323 operation mode to phone to phone (P2P), not gatekeeper (GK):

```
usr/config$ h323 -print
H.323 stack relate information
    RAS mode               : Non-GK mode
    Registered e164        : 31
    Registered H323 ID     : Rob
    RTP port               : 16384
    H.245 port             : 16640
    Allocated port range   :
              start port   : 1024
              end port     : 65535
    Response timeOut       : 5
    Connect  timeOut       : 5000
usr/config$
```

4. Add the gateway's address to the phonebook:

```
usr/config$ pbook -add name gw ip 10.1.1.12
usr/config$
This may take a few seconds, please wait....

Commit to flash memory ok!

usr/config$ pbook -print
index   Name                 IP                    E164
=========================================================================
1       gw                   10.1.1.12
-------------------------------------------------------------------------
usr/config$
```

Making calls from the IP telephone 10.5.8.2:

- Just lift the handset and dial '11', or '13' fo the PBX extensions.
- Dial '33' for [Joe]. The call request will be sent to the gateway 10.1.1.12, where it will be forwarded to [Joe]. If you want to call [Joe] directly, add a phonebook record for it:
  `usr/config$ pbook -add name Joe ip 10.0.0.224 e164 33`

Use the telephony logging feature on the gateway to debug your setup.

Setting up MikroTik Router and CISCO Router

Let's try a different example.

Here are some hints on how to get working configuration for telephony calls between CISCO and MikroTik router.

Configuration on the MikroTik side

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- G.729a codec MUST be disabled (otherwise connections are not possible at all!!!)
  `/ip telephony codec disable G.729A-8k/sw`
- G.711-ALaw codec should not be used (in some cases there is no sound)
  `/ip telephony codec disable "G.711-ALaw-64k/sw G.711-ALaw-64k/hw"`
- Fast start has to be used (otherwise no ring-back tone and problems with codec negotiation)
  `/ip telephony voice-port set cisco fast-start=yes`
- Telephone number we want to call to must be sent to Cisco, for example
  `/ip telephony numbers add destination-pattern=101 voice-port=cisco prefix=101`
- Telephone number, cisco will call us, must be assigned to some voice port, for example,
  `/ip telephony numbers add destination-pattern=098 voice-port=linejack`

Configuration on the CISCO side:

- IP routing has to be enabled
  `ip routing`
- Default values for fast start can be used:
  `voice service pots default h323 call start exit voice service voip default h323 call start exit`
- Enable opening of RTP streams:
  `voice rtp send-recv`
- Assign some E.164 number for local telephone, for example, 101 to port 0/0
  `dial-peer voice 1 pots destination-pattern 101 port 0/0 exit`
- create preferred codec listing:
  `voice class codec codec_class_number codec preference 1 g711ulaw codec preference 2 g723r63 exit`
  NOTE: g723r53 codec can be used, too
- Tell, that some foreign E.164 telephone number can be reached by calling to some IP address, for example, 098 by calling to 10.0.0.98
  `dial-peer voice 11 voip destination-pattern 098 session target ipv4:10.0.0.98 voice-class codec codec_class_number exit`
  NOTE: instead of codec class, one specified codec could be specified:
  `codec g711ulaw`

For reference, following is an exported CISCO configuration, that works:

```
!
version 12.1
no service single-slot-reload-enable
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
!
logging rate-limit console 10 except errors
enable secret 5 $1$bTMC$nDGl9/n/pc3OMbtWxADMg1
enable password 123
!
memory-size iomem 25
ip subnet-zero
no ip finger
!
```

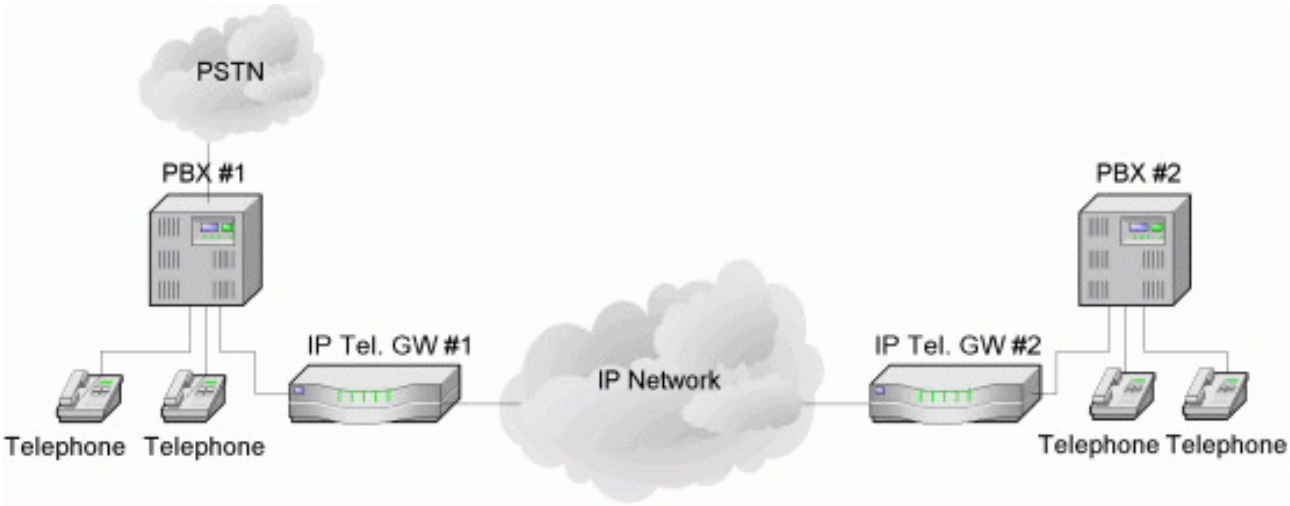

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
call rsvp-sync
voice rtp send-recv
!
voice class codec 1
 codec preference 1 g711ulaw
 codec preference 2 g723r63
!
interface FastEthernet0
 ip address 10.0.0.101 255.255.255.0
 no ip mroute-cache
 speed auto
 half-duplex
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.0.0.1
no ip http server
!
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipx permit
!
voice-port 0/0
!
voice-port 0/1
!
voice-port 2/0
!
voice-port 2/1
!
dial-peer voice 1 pots
 destination-pattern 101
 port 0/0
!
dial-peer voice 97 voip
 destination-pattern 097
 session target ipv4:10.0.0.97
 codec g711ulaw
!
dial-peer voice 98 voip
 destination-pattern 098
 voice-class codec 1
 session target ipv4:10.0.0.98
!
!
line con 0
 transport input none
line aux 0
line vty 0 4
 password 123
 login
!
end
```

Setting up PBX to PBX Connection over an IP Network

To interconnect two telephone switchboards (PBX) over an IP network, two IP telephony gateways should be configured. The setup is shown in the following diagram:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.



We want to be able to use make calls from local telephones of one PBX to local telephones or external lines of the other PBX.

Assume that:

- The IP telephony gateway #1 has IP address 10.0.0.182, and the name of the Voicetronix first line is 'vctx1'.
- The IP telephony gateway #2 has IP address 10.0.0.183, and the name of the Voicetronix first line is 'vctx1'.

The IP telephony configuration should be as follows:

- IP telephony gateway #1 should have:
```
/ip telephony voice-port voip add name=gw2 remote-address=10.0.0.183 /ip telephony
numbers add dst-pattern=1.. voice-port=gw2 prefix=2 add dst-pattern=2..
voice-port=vctx1 prefix=1
```
- IP telephony gateway #2 should have
```
/ip telephony voice-port voip add name=gw1 remote-address=10.0.0.182 /ip telephony
numbers add dst-pattern=2.. voice-port=vctx1 prefix=1 add dst-pattern=1..
voice-port=gw1 prefix=2
```

The system works as follows:

To dial from the main office PBX#1 any extension of the remote office PBX#2, the extension with the connected gateway at PBX#1 should be dialed first. Then, after the dial tone of the gateway#1 is received, the remote extension number should be dialed.

To dial from the main office PBX#2 any extension of the remote office PBX#1, the actions are the same as in first situation.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

System Watchdog

Document revision 1.2 (Tue Mar 09 08:45:49 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
Hardware Watchdog Management
  Description
  Property Description
  Example

General Information

Summary

System watchdog feature is needed to reboot the system in case of software failures.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/system watchdog***
Hardware usage: ***Not significant***

Hardware Watchdog Management

Home menu level: ***/system watchdog***

Description

This menu allows to configure system to reboot on kernel panic, when an IP address does not respond, or in case the system has locked up. Software watchdog timer is used to provide the last option, so in very rare cases (caused by hardware malfunction) it can lock up by itself. There is a hardware watchdog device available in RouterBOARD hardware, which can reboot the system in any case.

Property Description

reboot-on-failure (yes | no; default: **no**) - whether to reboot on kernel panic

watch-address (*IP address*; default: **none**) - if set, the system will reboot in case 6 sequental pings to the given IP address (sent once per 10 seconds) will fail

- **none** - disable this option

watchdog-timer (yes | no; default: **no**) - whether to reboot if system is unresponsive for a minute

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

no-ping-delay (*time*; default: **5m**) - specifies how long after reboot not to test and ping watch-address. The default setting means that if watch-address is set and is not reachable, the router will reboot about every 6 minutes.

automatic-supout (yes | no; default: **yes**) - when software failure happens, a file named "autosupout.rif" is generated automatically. The previous "autosupout.rif" file is renamed to "autosupout.old.rif"

auto-send-supout (yes | no; default: **no**) - after the support output file is automatically generated, it can be sent by email

send-email-from (*text*; default: **""**) - e-mail address to send the support output file from. If not set, the value set in /tool e-mail is used

send-email-to (*text*; default: **""**) - e-mail address to send the support output file to

send-smtp-server (*text*; default: **""**) - SMTP server address to send the support output file through. If not set, the value set in /tool e-mail is used

Example

To make system generate a support output file and sent it automatically to **support@example.com** throught the **192.0.2.1**in case of a software crash:

```
[admin@MikroTik] system watchdog> set auto-send-supout=yes \
\... send-to-email=support@example.com send-smtp-server=192.0.2.1
[admin@MikroTik] system watchdog> print
  reboot-on-failure: yes
      watch-address: none
     watchdog-timer: yes
      no-ping-delay: 5m
   automatic-supout: yes
   auto-send-supout: yes
   send-smtp-server: 192.0.2.1
      send-email-to: support@example.com
[admin@MikroTik] system watchdog>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

UPS Monitor

Document revision 2.2 (Thu Jul 07 17:18:54 GMT 2005)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
  Summary
  Specifications
  Related Documents
  Description
UPS Monitor Setup
  Property Description
  Notes
  Example
Runtime Calibration
  Description
  Notes
  Example
UPS Monitoring
  Property Description
  Example

General Information

Summary

The UPS monitor feature works with APC UPS units that support “smart” signaling over serial RS232 or USB connection. This feature enables the network administrator to monitor the UPS and set the router to ‘gracefully’ handle any power outage with no corruption or damage to the router. The basic purpose of this feature is to ensure that the router will come back online after an extended power failure. To do this, the router will monitor the UPS and set itself to hibernate mode when the utility power is down and the UPS battery is has less than 10% of its battery power left. The router will then continue to monitor the UPS (while in hibernate mode) and then restart itself after when the utility power returns. If the UPS battery is drained and the router loses all power, the router will power back to full operation when the ‘utility’ power returns.

The UPS monitor feature on the MikroTik RouterOS supports

- hibernate and safe reboot on power and battery failure
- UPS battery test and run time calibration test
- monitoring of all "smart" mode status information supported by UPS
- logging of power changes

Specifications

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Packages required: ***ups***
License required: ***level1***
Home menu level: ***/system ups***
Standards and Technologies: ***APC's smart protocol***
Hardware usage: ***Not significant***

Related Documents

- ***Software Package Management***

Description

Cabling

The APC UPS (BackUPS Pro or SmartUPS) requires a special serial cable. If no cable came with the UPS, a cable may be ordered from APC or one can be made "in-house". Use the following diagram:

Router Side (DB9f)	Signal	Direction	UPS Side (DB9m)
2	**Receive**	**IN**	**2**
3	**Send**	**OUT**	**1**
5	**Ground**		**4**
7	**CTS**	**IN**	**6**

Note that you may also connect with USB if available.

UPS Monitor Setup

Home menu level: ***/system ups***

Property Description

alarm-setting (*delayed* | *immediate* | *low-battery* | *none*; default: **immediate**) - UPS sound alarm setting:

- **delayed** - alarm is delayed to the on-battery event
- **immediate** - alarm immediately after the on-battery event
- **low-battery** - alarm only when the battery is low
- **none** - do not alarm

load (*read-only: percentage*) - the UPS's output load as a percentage of full rated load in Watts. The typical accuracy of this measurement is ±3% of the maximum of 105%

manufacture-date (*read-only: text*) - the UPS's date of manufacture in the format "mm/dd/yy" (month, day, year)

min-runtime (*time*; default: **5m**) - minimal run time remaining. After a 'utility' failure, the router will monitor the runtime-left value. When the value reaches the min-runtime value, the router will go to hibernate mode

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- **0** - the router will go to hibernate mode when the "battery low" signal is sent indicating that the battery power is below 10%

model (*read-only: text*) - less than 32 ASCII character string consisting of the UPS model name (the words on the front of the UPS itself)

nominal-battery-voltage (*read-only: integer*) - the UPS's nominal battery voltage rating (this is not the UPS's actual battery voltage)

offline-time (*time*; default: **5m**) - how long to work on batteries. The router waits that amount of time and then goes into hibernate mode until the UPS reports that the 'utility' power is back

- **0** - the router will go into hibernate mode according the min-runtime setting and 10% of battery power event. In this case, the router will wait until the UPS reports that the battery power is below 10%

port (*name*) - communication port of the router

serial (*read-only: text*) - a string of at least 8 characters directly representing the UPS's serial number as set at the factory. Newer SmartUPS models have 12-character serial numbers

version (*read-only: text*) - UPS version, consists of three fields: SKU number, firmware revision, country code. The county code may be one of the following:

- **I** - 220/230/240 Vac
- **D** - 115/120 Vac
- **A** - 100 Vac
- **M** - 208 Vac
- **J** - 200 Vac

Notes

In order to enable UPS monitor, the serial port should be available.

Example

To enable the UPS monitor for port **serial1**:

```
[admin@MikroTik] system ups> add port=serial1 disabled=no
[admin@MikroTik] system ups> print
Flags: X - disabled, I - invalid
 0    name="ups" port=serial1 offline-time=5m min-runtime=5m
      alarm-setting=immediate model="SMART-UPS 1000" version="60.11.I"
      serial="QS0030311640" manufacture-date="07/18/00"
      nominal-battery-voltage=24V
[admin@MikroTik] system ups>
```

Runtime Calibration

Command name: ***/system ups rtc***

Description

The **rtc** command causes the UPS to start a run time calibration until less than 25% of full battery capacity is reached. This command calibrates the returned run time value.

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

Notes

The test begins only if the battery capacity is 100%.

Example

```
[admin@MikroTik] system ups> rtc 0
```

UPS Monitoring

Command name: ***/system ups monitor***

Property Description

battery-charge (*percentage*) - the UPS's remaining battery capacity as a percent of the fully charged condition

battery-voltage - the UPS's present battery voltage. The typical accuracy of this measurement is ±5% of the maximum value (depending on the UPS's nominal battery voltage)

frequency (*percentage*) - when operating on-line, the UPS's internal operating frequency is synchronized to the line within variations within 3 Hz of the nominal 50 or 60 Hz. The typical accuracy of this measurement is ±1% of the full scale value of 63 Hz

line-voltage - the in-line utility power voltage

load (*percentage*) - the UPS's output load as a percentage of full rated load in Watts. The typical accuracy of this measurement is ±3% of the maximum of 105%

low-battery - only shown when the UPS reports this status

on-battery (*yes* | *no*) - Whether UPS battery is supplying power

on-line (*yes* | *no*) - whether power is being provided by the external utility (power company)

output-voltage - the UPS's output voltage

overloaded-output - only shown when the UPS reports this status

replace-battery - only shown when the UPS reports this status

runtime-calibration-running - only shown when the UPS reports this status

runtime-left (*time*) - the UPS's estimated remaining run time in minutes. You can query the UPS when it is operating in the on-line, bypass, or on-battery modes of operation. The UPS's remaining run time reply is based on available battery capacity and output load

smart-boost-mode - only shown when the UPS reports this status

smart-ssdd-mode - only shown when the UPS reports this status

transfer-cause (*text*) - the reason for the most recent transfer to on-battery operation (only shown when the unit is on-battery)

Example

When running on utility power:

```
[admin@MikroTik] system ups> monitor 0
          on-line: yes
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
        on-battery: no
       RTC-running: no
      runtime-left: 20m
    battery-charge: 100%
   battery-voltage: 27V
      line-voltage: 226V
    output-voltage: 226V
              load: 45%
       temperature: 39C
         frequency: 50Hz
   replace-battery: no
       smart-boost: no
        smart-trim: no
          overload: no
       low-battery: no

 [admin@MikroTik] system ups>
```

When running on battery:

```
 [admin@MikroTik] system ups> monitor 0
           on-line: no
         on-battery: yes
    transfer-cause: "Line voltage notch or spike"
       RTC-running: no
      runtime-left: 19m
     offline-after: 4m46s
    battery-charge: 94%
   battery-voltage: 24V
      line-voltage: 0V
    output-voltage: 228V
              load: 42%
       temperature: 39C
         frequency: 50Hz
   replace-battery: no
       smart-boost: no
        smart-trim: no
          overload: no
       low-battery: no

       [admin@MikroTik] system ups>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

VRRP

Document revision 1.4 (Fri Mar 05 08:42:58 GMT 2004)
This document applies to MikroTik RouterOS V2.8

Table of Contents

Table of Contents
General Information
  Summary
  Specifications
  Related Documents
  Description
VRRP Routers
  Description
  Property Description
  Notes
Virtual IP addresses
  Property Description
  Notes
A simple example of VRRP fail over
  Description
  Configuring Master VRRP router
  Configuring Backup VRRP router
  Testing fail over

General Information

Summary

Virtual Router Redundancy Protocol (VRRP) implementation in the MikroTik RouterOS is RFC2338 compliant. VRRP protocol is used to ensure constant access to some resources. Two or more routers (referred as VRRP Routers in this context) create a highly available cluster (also referred as Virtual routers) with dynamic fail over. Each router can participate in not more than 255 virtual routers per interface. Many modern routers support this protocol.

Network setups with VRRP clusters provide high availability for routers without using clumsy ping-based scripts.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip vrrp***
Standards and Technologies: ***VRRP***, ***AH***, ***HMAC-MD5-96 within ESP and AH***
Hardware usage: ***Not significant***

Related Documents

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

- ***Package Management***
- ***IP Addresses and ARP***

Description

Virtual Router Redundancy Protocol is an election protocol that provides high availability for routers. A number of routers may participate in one or more virtual routers. One or more IP addresses may be assigned to a virtual router. A node of a virtual router can be in one of the following states:

- **MASTER** state, when the node answers all the requests to the instance's IP addresses. There may only be one MASTER node in a virtual router. This node sends VRRP advertisement packets to all the backup routers (using multicast address) every once in a while (set in **interval** property).
- **BACKUP** state, when the VRRP router monitors the availability and state of the Master Router. It does not answer any requests to the instance's IP addresses. Should master become unavailable (if at least three sequential VRRP packets are lost), election process happens, and new master is proclaimed based on its priority. For more details on virtual routers, see RFC2338.

VRRP Routers

Home menu level: ***/ip vrrp***

Description

A number of VRRP routers may form a virtual router. The maximal number of clusters on one network is 255 each having a unique VRID (Virtual Router ID). Each router participating in a VRRP cluster must have it priority set to a valid value.

Property Description

name (*name*) - assigned name of the VRRP instance

interface (*name*) - interface name the instance is running on

vrid (*integer*: 0..255; default: **1**) - Virtual Router Identifier (must be unique on one interface)

priority (*integer*: 1..255; default: **100**) - priority of the current node (higher values mean higher priority)

- **255** - RFC requires that the router that owns the IP addresses assigned to this instance had the priority of 255

interval (*integer*: 1..255; default: **1**) - VRRP update interval in seconds. Defines how frequently the master of the given cluster sends VRRP advertisement packets

preemption-mode (yes | no; default: **yes**) - whether preemption mode is enabled

- **no** - a backup node will not be elected to be a master until the current master fail even if the backup node has higher priority than the current master
- **yes** - the master node always has the priority

authentication (*none* | *simple* | *ah*; default: **none**) - authentication method to use for VRRP

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

advertisement packets

- **none** - no authentication
- **simple** - plain text authentication
- **ah** - Authentication Header using HMAC-MD5-96 algorithm

password (*text*; default: "") - password required for authentication depending on method used can be ignored (if no authentication used), 8-character long text string (for plain-text authentication) or 16-character long text string (128-bit key required for AH authentication)

on-backup (*name*; default: "") - script to execute when the node switch to backup state

on-master (*name*; default: "") - script to execute when the node switch to master state

Notes

All the nodes of one cluster must have the same **vrid**, **interval**, **preemption-mode**, **authentication** and **password**.

As said before, priority of 255 is reserved for the real owner of the virtual router's IP addresses. Theoretically, the owner should have the IP address added statically to its IP address list and also to the VRRP virtual address list, but you should never do this! Any addresses that you are using as virtual addresses (i.e. they are added in **/ip vrrp address**) must not appear in **/ip address** list as they otherwise can cause IP address conflict, which will not be resolved automatically.

Also You must have an IP address (no matter what) on the interface you want to run VRRP on.

Example

To add a VRRP instance on **ether1** interface, forming (because **priority** is **255**) a virtual router with **vrid** of **1**:

```
[admin@MikroTik] ip vrrp> add interface=ether1 vrid=1 priority=255
[admin@MikroTik] ip vrrp> print
Flags: X - disabled, I - invalid, M - master, B - backup
  0  I  name="vr1" interface=ether1 vrid=1 priority=255 interval=1
        preemption-mode=yes authentication=none password="" on-backup=""
        on-master=""

[admin@MikroTik] ip vrrp>
```

Virtual IP addresses

Home menu level: ***/ip vrrp address***

Property Description

address (*IP address*) - IP address belongs to the virtual router

network (*IP address*) - IP address of the network

broadcast (*IP address*) - broadcasting IP address

virtual-router (*name*) - VRRP router's name the address belongs to

Notes

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

The virtual IP addresses should be the same for each node of a virtual router.

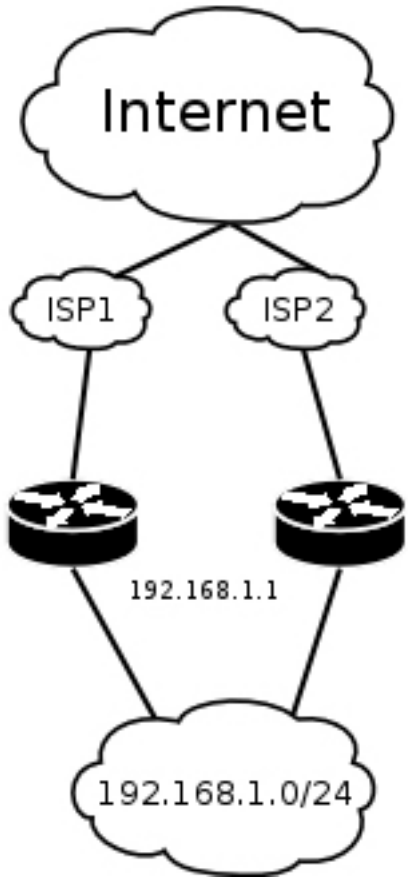
Example

To add a virtual address of **192.168.1.1/24** to the **vr1** VRRP router:

```
[admin@MikroTik] ip vrrp> address add address=192.168.1.1/24 \
\... virtual-router=vr1
[admin@MikroTik] ip vrrp> address print
Flags: X - disabled, A - active
  #    ADDRESS            NETWORK         BROADCAST       VIRUAL-ROUTER
  0    192.168.1.1/24     192.168.1.0     192.168.1.255   vr1

[admin@MikroTik] ip vrrp>
```

A simple example of VRRP fail over

Description



VRRP protocol may be used to make a redundant Internet connection with seamless fail-over. Let us assume that we have 192.168.1.0/24 network and we need to provide highly available Internet connection for it. This network should be NATted (to make fail-over with public IPs, use such dynamic routing protocols as BGP or OSPF together with VRRP). We have connections to two different Internet Service Providers (ISPs), and one of them is preferred (for example, it is cheaper or faster).

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

This example shows how to configure VRRP on the two routers shown on the diagram. The routers must have initial configuration: interfaces are enabled, each interface have appropriate IP address (note that each of the two interfaces should have an IP address), routing table is set correctly (it should have at least a default route). SRC-NAT or masquerading should also be configured before. See the respective manual chapters on how to make this configuration.

We will assume that the interface the 192.168.1.0/24 network is connected to is named **local** on both VRRP routers

Configuring Master VRRP router

First of all we should create a VRRP instance on this router. We will use the priority of 255 for this router as it should be preferred router.

```
[admin@MikroTik] ip vrrp> add interface=local priority=255
[admin@MikroTik] ip vrrp> print
Flags: X - disabled, I - invalid, M - master, B - backup
  0   M name="vr1" interface=local vrid=1 priority=255 interval=1
        preemption-mode=yes authentication=none password="" on-backup=""
        on-master=""

[admin@MikroTik] ip vrrp>
```

Next the virtual IP address should be added to this VRRP instance

```
[admin@MikroTik] ip vrrp> address add address=192.168.1.1/24 \
\... virtual-router=vr1
[admin@MikroTik] ip vrrp> address print
Flags: X - disabled, A - active
  #    ADDRESS            NETWORK         BROADCAST       VIRTUAL-ROUTER
  0    192.168.1.1/24     192.168.1.0     192.168.1.255   vr1

[admin@MikroTik] ip vrrp>
```

Now this address should appear in **/ip address** list:

```
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.0.0.1/24        10.0.0.0        10.0.0.255      public
  1   192.168.1.2/24     192.168.1.0     192.168.1.255   local
  2 D 192.168.1.1/24     192.168.1.0     192.168.1.255   local

[admin@MikroTik] ip address>
```

Configuring Backup VRRP router

Now we will create VRRP instance with lower priority (we can use the default value of **100**), so this router will back up the preferred one:

```
[admin@MikroTik] ip vrrp> add interface=local
[admin@MikroTik] ip vrrp> print
Flags: X - disabled, I - invalid, M - master, B - backup
  0   B name="vr1" interface=local vrid=1 priority=100 interval=1
        preemption-mode=yes authentication=none password="" on-backup=""
        on-master=""

[admin@MikroTik] ip vrrp>
```

Now we should add the same virtual address as was added to the master node:

Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.

```
[admin@MikroTik] ip vrrp> address add address=192.168.1.1/24 \
\... virtual-router=vr1
[admin@MikroTik] ip vrrp> address print
Flags: X - disabled, A - active
  #    ADDRESS            NETWORK         BROADCAST       VIRTUAL-ROUTER
  0    192.168.1.1/24     192.168.1.0     192.168.1.255   vr1

[admin@MikroTik] ip vrrp>
```

Note that this address will not appear in **/ip address** list:

```
[admin@MikroTik] ip address> print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.1.0.1/24        10.0.0.0        10.0.0.255      public
  1   192.168.1.3/24     192.168.1.0     192.168.1.255   local

[admin@MikroTik] ip address>
```

Testing fail over

Now, when we will disconnect the master router, the backup one will switch to the master state:

```
[admin@MikroTik] ip vrrp> print
Flags: X - disabled, I - invalid, M - master, B - backup
  0   M name="vr1" interface=local vrid=1 priority=100 interval=1
        preemption-mode=yes authentication=none password="" on-backup=""
        on-master=""

[admin@MikroTik] ip vrrp> /ip address print
Flags: X - disabled, I - invalid, D - dynamic
  #   ADDRESS            NETWORK         BROADCAST       INTERFACE
  0   10.1.0.1/24        10.0.0.0        10.0.0.255      public
  1   192.168.1.3/24     192.168.1.0     192.168.1.255   local
  2 D 192.168.1.1/24     192.168.1.0     192.168.1.255   local

[admin@MikroTik] ip vrrp>
```


Copyright 1999-2005, MikroTik. All rights reserved. Mikrotik, RouterOS and RouterBOARD are trademarks of Mikrotikls SIA.
Other trademarks and registred trademarks mentioned herein are properties of their respective owners.